



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DOTTORATO DI RICERCA IN
COMPUTER SCIENCE AND ENGINEERING

Ciclo 37

Settore Concorsuale: 01/B1 - INFORMATICA

Settore Scientifico Disciplinare: INF/01 - INFORMATICA

HARMONISING MUSIC INFORMATION RETRIEVAL WITH SEMANTICS: FROM
DATA INTEGRATION TO MULTIMODALITY

Presentata da: Andrea Poltronieri

Coordinatore Dottorato

Ilaria Bartolini

Supervisore

Valentina Presutti

Esame finale anno 2025

“Science cannot tell us a word about why music delights us, of why and how an old song can move us to tears. Science can, in principle, describe in full detail all that happens in the latter case in our sensorium and ‘motorium’ from the moment the waves of compression and dilation reach our ear to the moment when certain glands secrete a salty fluid that emerges from our eyes.”

Erwin Schrödinger

Abstract

IN the era of big data and machine learning, the fragmentation of musical datasets and the lack of standardised representations continue to hinder advancements in Music Information Retrieval (MIR). The multifaceted nature of music complicates both the representation of *content*—with small, task-specific datasets scattered across various formats, and *context* (metadata), where there is a lack of a consistent, structured terminology. These challenges increase the effort required for data collection and pre-processing, reduce reproducibility, and limit the scalability of MIR models. To address these issues, this thesis proposes a unified semantic model to foster interoperability and advance MIR tasks.

A specific instance of this fragmentation can be found in harmonic annotations, where harmony—an essential musical dimension—is inconsistently represented across datasets, formats, and notational systems. Taking harmony as a use case and leveraging the proposed semantic model, this thesis develops a standardised workflow to harmonise previously disconnected datasets, enabling the creation of new, large-scale unified corpora. Building on these harmonised datasets, a key contribution of the thesis is the exploration of harmonic similarity, which is used to organise, explore, and reveal connections across diverse tracks, historical periods, and genres. To this end, we implement novel state-of-the-art harmonic similarity functions, advancing current research in MIR. Moreover, by utilising the created large, varied data collections, we uncover deeper relationships within music and explore how diverse stylistic excerpts can be used for generative tasks.

While integrating symbolic data offers significant advantages, certain limita-

tions persist. Primary challenges include the limited diversity of annotated data, often biased toward a narrow range of musical genres, and the inherent ambiguity and subjectivity in harmonic annotations. Such challenges have led MIR tasks like Audio Chord Estimation (ACE) to hit a “glass ceiling,” where neither increasing computational power nor the volume of data has led to improved results.

To address these issues, this thesis explores a multimodal approach aimed at enhancing the analysis and understanding of harmonic data by jointly leveraging audio and chord annotations. We first propose a novel method for enriching the dataset with audio annotations aligned to the existing symbolic data. Building on this foundation, we introduce a new model for ACE that embeds formalised music theory concepts such as consonance and dissonance, addressing both training and evaluation challenges in the field. This model aims to mitigate the limitations of chord vocabulary imbalance and annotation subjectivity, ultimately improving the state-of-the-art in audio-based harmonic analysis.

Contents

Abstract	II
1 Introduction	1
1.1 Problem Definition and Research Questions	2
1.1.1 Challenges in Representing Music: The Need for Unified Semantic Models	2
1.1.2 Harmonising Symbolic Data: Addressing Fragmentation in Harmonic Datasets	6
1.1.3 Exploring Harmonic Similarity: Leveraging Large-Scale Corpora for Deeper Musical Understanding	7
1.1.4 Limitations of Symbolic Data Integration: A Multimodal Approach	9
1.1.5 Limitations in Audio Chord Estimation: Chord Imbalance and Subjectivity	11
1.2 Thesis Contribution	13
1.3 Thesis Structure	17
2 Background	21
2.1 Music Theory and Structure	22
2.1.1 Fundamentals of Music Theory	23
2.1.2 Facets of Music	30
2.1.3 Focus on Harmony	34
2.2 Music Technology	39
2.2.1 Historical Background	39
2.2.2 Music Information Retrieval Tasks	40
2.2.3 Harmony in MIR	42
2.3 Music Representation	43
2.3.1 Representing Musical Metadata	44
2.3.2 Representing Musical Content	46

2.3.3	Audio Representation of Music	49
2.3.4	Symbolic Representation of Music	52
2.3.5	Knowledge Representation of Music	58
2.4	Multimodality in Music Information Retrieval	61
2.4.1	Towards a Definition of Multimodality in MIR	62
2.4.2	Challenges of Multimodal Deep Learning for MIR	64
3	Representing Musical Knowledge	67
3.1	Introduction	67
3.1.1	Challenges and Requirements for Interoperability	68
3.1.2	Towards a Unified Model for Music Representation	71
3.1.3	Our contribution	72
3.1.4	Chapter Structure	73
3.2	Related Work	74
3.2.1	Ontologies for Describing Music Context	74
3.2.2	Ontologies for Describing Music Content	75
3.2.3	Ontology Engineering Methodologies	78
3.3	The eXtreme Design Methodology in Polifonia	79
3.3.1	Requirements collection	79
3.3.2	Ontology Network Design and Development	81
3.4	The Polifonia Ontology Network (PON)	83
3.4.1	Foundational models and their extensions and specialisations	84
3.4.2	Modules for analysis and annotation of music	89
3.5	The Music Meta Ontology	90
3.5.1	Main elements of design	91
3.5.2	Conversion rules and code support	95
3.6	Adoption and impact	96
3.6.1	Current use by Polifonia pilots	96
3.6.2	Survey of interest for future applications	97
3.6.3	Adoption by Polifonia Stakeholders	99
3.6.4	Availability, sustainability, and FAIRness	99
3.7	Conclusions	99
4	Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration	103
4.1	Introduction	103

Contents

4.1.1	Our contribution	105
4.1.2	Chapter Structure	107
4.2	Related Work	107
4.3	ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs	109
4.3.1	Methods	109
4.3.2	Data Records	126
4.4	Technical Validation	132
4.4.1	Validation of the JAMifier	132
4.4.2	Validation of the Chonverter	135
4.5	Usage Notes	136
4.5.1	Applications and tasks	137
4.5.2	Online survey	139
4.6	Data Availability	141
4.7	Conclusion	142
5	Uncovering Harmonic Similarity: From Musicological to Creative Exploration	145
5.1	Introduction	145
5.1.1	Harmonic Similarity for Musicological Exploration	147
5.1.2	Supporting Music Creativity	147
5.1.3	Our contribution	149
5.1.4	Chapter Structure	151
5.2	Related Work	151
5.2.1	Content-based Similarity in the Symbolic Domain	151
5.2.2	Harmonic Similarity in the Symbolic Domain	152
5.2.3	Computational Models for assisting Creativity	153
5.3	LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures	154
5.3.1	Encoding of symbolic chord sequences	154
5.3.2	Pattern extraction and matching	156
5.3.3	Preliminary experiments	158
5.3.4	Analysis of genre-specific harmonic dependencies	159
5.3.5	The interactive harmonic network	162
5.4	Harmory: The Harmonic Memory	164
5.4.1	Knowledge graph creation	169
5.4.2	Experiments	170

5.4.3	Avenues for machine creativity	175
5.5	Conclusion	178
5.5.1	Limitations and Future Work	179
6	Exploring Symbolic Limitations: Multimodal Strategies for Enhanced Harmonic Analysis	181
6.1	Introduction	181
6.1.1	Limits and Challenges to Harmonic Data Integration	182
6.1.2	The Need for Multimodality	184
6.1.3	Our Contribution	187
6.1.4	Chapter Structure	188
6.2	Related Work	189
6.2.1	Alignment Techniques	189
6.2.2	Audio Chord Estimation (ACE)	190
6.2.3	Conformer-based Approaches	192
6.3	ChordSync: Conformer-Based Alignment of Chord Annotations to Music Audio	192
6.3.1	Problem Statement	193
6.3.2	Preprocessing	195
6.3.3	Conformer-based Acoustic Model	195
6.3.4	Evaluation	198
6.4	Inter-annotator Agreement Analysis	201
6.5	From Dissonance to Harmony	204
6.5.1	Methods	204
6.5.2	Evaluation	207
6.6	Conclusion	210
6.6.1	Limitations and Future Work	211
7	Conclusion	215
7.1	Summary of Contributions	216
7.2	Discussion and Future Work	219
7.2.1	Ontology Engineering and Data Integration	219
7.2.2	Dataset Expansion and Workflow Adaptation	220
7.2.3	Symbolic Harmonic Similarity and Exploration	220
7.2.4	Multimodality	221
	Acknowledgments	225

Contents

List of Figures	231
List of Tables	234
List of Abbreviations	237
Bibliography	242

CHAPTER *1*

Introduction

The intersection of music and computer science has changed the way we analyse, understand, and interact with music. Over the past decades, advancements in computational techniques have provided researchers with powerful tools for exploring large-scale music collections and analysing the structure of music in unprecedented ways. These tools have enabled the processing of vast amounts of musical data – far exceeding human capabilities – enabling novel approaches in systematic musicology, music discovery, and computer-assisted composition.

This transformation began in the 1950s, when the first computational applications to music were conceptualised and gradually implemented during the 1960s [30]. At that time, computational models supported researchers in challenges that traditional methods could not manage, such as identifying patterns across hundreds of compositions, conducting statistical analyses, and developing formalised representations of music. By the late 20th century, the digitisation of music and the rapid growth of digital archives accelerated the demand for intelligent systems capable of managing and retrieving musical information. This development laid the foundation for Music Information Retrieval (MIR), a field focused on creating

models for handling and analysing vast digital music collections.

As music archives expanded in size and complexity, new challenges emerged around how to represent and access different aspects of music. MIR, which established itself as a formal research field in the early 2000s [120], focuses on addressing these challenges by developing techniques for extracting meaningful information from diverse music data representations, such as audio recordings, symbolic data, or music metadata. Central tasks such as *retrieval*, *recommendation*, and *browsing* have become core paradigms of the field, enhancing user interactions and facilitating seamless access to music collections [221].

Despite these advancements, a consistent and meaningful representation of both musical content and metadata remains an unresolved problem due to the complex nature of music itself. Music consists of a variety of structural elements, such as melody, harmony, and rhythm, that can be represented differently in terms of notation and conceptual model, depending on the task to be addressed or the available dataset(s). In addition, music is deeply embedded in diverse cultural and historical contexts, adding further complexity to the metadata that any retrieval or analysis system should account for [41]. This complexity has led to the fragmentation of music data into isolated, task-specific formats, notations, and conceptual models, making it increasingly difficult to develop unified frameworks that scale across diverse datasets, styles and genres. We argue that addressing this fragmentation by establishing standardised approaches to music representation is essential for advancing MIR models and improving the performances on a broad range of MIR tasks.

1.1 Problem Definition and Research Questions

1.1.1 Challenges in Representing Music: The Need for Unified Semantic Models

The challenge of understanding and processing the multiple dimensions of music within MIR is often referred to as the “multifaceted challenge” and these facets are categorised by Downie [119] into seven distinct categories: *pitch*, *temporal*, *harmonic*, *timbral*, *editorial*, *textual*, and *bibliographic*.

While the first six facets primarily address music content, encapsulating the diverse elements that make up a musical piece, the bibliographic facet extends beyond the musical content itself, capturing broader contextual information, such

1.1. Problem Definition and Research Questions

as the historical, cultural, and publication data surrounding a musical work. This facet is also known as *knowledge about music* [296], or more commonly, *music metadata*. The distinction between content and context forms the basis of two major paradigms within MIR: *Content-based MIR*, which deals with the intrinsic properties of music, usually further categorised into *signal* and *symbolic* representations, and *Context-based MIR*, which focuses on external, contextual information. Both paradigms are widely used for tasks such as music similarity, recommendation, and browsing, either in isolation or in combination [221].

Effectively representing both the content and context of music presents one of the central challenges within MIR. Capturing the context, such as cultural or historical information, is particularly complex. A significant part of the difficulty stems from the lack of a consistent and universally accepted terminology for describing music metadata. For instance, in conventional databases, the “artist” is typically the central entity, contrasting with the emphasis on the “composer” in Western classical music and with the concept of “performer” in Western popular music. Similarly, the concept of a “composition” in the classical tradition diverges significantly from terms like “track” or “song,” which dominate in most contemporary systems. These problems have repercussions on many MIR tasks, both when searching for data in a database and when integrating data from different datasets.

Furthermore, musical heritage encompasses a diversity of human expressions and experiences across different cultural and historical contexts. This diversity is reflected in the metadata, where multiple sources (e.g., theatres, conservatoires, instruments) must be connected to their broader social and artistic contexts (e.g., scholars, musicians, intellectuals) across different languages and time periods [41]. These issues are exacerbated when the context includes conflicting or incomplete information, such as varying names for the same artist across genres or historical periods. For instance, music metadata often lacks a unified structure to account for oral traditions (common in folk music), where works evolve through verbal transmission rather than being formally composed and notated.

Semantic models have been proposed to standardise the description of music metadata (e.g. [324, 64]), however, these models are mostly limited in scope. They focus on specific historical periods or genres and are tailored to narrow requirements, limiting their applicability to other styles or broader contexts. As a result, they are insufficient for generalising across diverse musical collections,

making it difficult to create a unified approach for organising and accessing music across varying genres and traditions.

The problem is further compounded when addressing music content. In fact, unlike other forms of knowledge, music lacks a consensual, shared meaning [296]. Whereas language or images usually point to specific objects or concepts, music consists of abstract elements—such as notes, chords, and sounds—that exist within their own domain, detached from external reference points. Historically, music notation has been the primary method for representing these abstract elements in the context of Western music. Originally developed to preserve the work over time and to allow performers to recreate the composer’s original intent [25], music notation has evolved over a thousand years [24], reflecting the growing complexity of Western music and its associated performance practices.

Computationally, music is typically represented in one of two ways: *signal representations* and *symbolic representations* [404]. *Signal representations* consist of recordings from sound sources. These representations are content-unaware and unstructured, making the extraction of information a complex task. In contrast, *symbolic representations* denote discrete musical events and provide a structured format that is context-aware, facilitating easier data extraction and analysis [396]. Depending on the task, different derivations of both audio representations (such as *MFCCs* or *Chroma Features*) and symbolic representations (such as *MIDI* or *MusicXML*) have been proposed [282]. This multiplicity of possible representations, known as the *Multirepresentational challenge* [119], has become increasingly significant. Each representation has unique implications for computational analysis: for instance, *MIDI* [106] representations capture performance details effectively but lack the expressive nuances of score-based formats, like *MEI* [174]. Selecting appropriate representations ensures that the data is both relevant and informative, which in turn enhances accuracy and reduces computational load [252].

The challenges of representing music content have led to a fragmentation in the field, where different datasets are represented using diverse symbolic representations, each tailored to serve specific tasks and applications. This fragmentation poses two major disadvantages: first, it requires significant resources for collecting, harmonising, and pre-processing disconnected and inconsistently represented data; second, it hampers the comparison and reproducibility of research results, as results derived from different datasets are not easily aligned [302]. As a consequence, the need for data interoperability has become increasingly critical. Data

1.1. Problem Definition and Research Questions

integration must address syntactic alignment (i.e., consistent data formats and structures) as well as semantic consistency, ensuring that the meaning and the relationships between musical elements are preserved across different representations. Achieving this requires data conversion as well as the formal modelling of musical information, such as by means of ontologies. Several ontologies have been developed to address various aspects of music, but they remain limited in scope. Most focus on modelling elements of music notation [332, 214, 315, 314], and some target specific formats, such as MIDI [272], or address a narrow set of audio signal features [7].

We argue that addressing the fragmentation of music data and the limitations of existing ontologies is essential for advancing MIR. This thesis proposes the development of a unified semantic model that integrates both music content and context, supporting interoperability and enhancing the reproducibility of MIR tasks. By streamlining data collection and pre-processing, such a model would improve the effectiveness of various MIR processes and provide a scalable, adaptable framework for handling diverse musical styles, genres, and formats. Furthermore, semantic models offer the advantage of continuous integration, as they can be extended and adapted to accommodate new, unforeseen use cases while maintaining interoperability. This adaptability ensures that the model remains relevant and scalable as the field evolves, supporting a wide range of tasks and musical data formats. This raises the first research question:

RQ1 *Is it possible to design an ontology representing both context- and content-based music information, by extending and generalizing over existing models? What features shall this ontology include in order to account the diversity of musical styles, genres, and formats?*

In addressing this fragmentation, we hypothesise that a unified semantic model can overcome limitations of existing ontologies by extending and generalising them across diverse musical domains. This model should aim to harmonise disparate data sources and provide a scalable, adaptable framework for supporting varied MIR tasks – thus setting the foundation for examining additional aspects of content- and context-based data integration.

1.1.2 Harmonising Symbolic Data: Addressing Fragmentation in Harmonic Datasets

A significant case where this challenge shows clearly is the case of harmonic annotations, where inconsistency across datasets, formats, and notations epitomises the broader issues within MIR. Harmony is a prominent dimension of music, also known as its “vertical dimension”. It is informally defined as “*combining notes in music to produce a pleasing effect greater than the sum of its parts*” [74]. Chords are the basic constituents of harmony, and sequences of them define the harmonic structure of a piece. A chord is defined as a simultaneous occurrence of several music sounds, producing harmony [165]. Depending on the notational system and the annotation conventions, a chord can be associated, for example, with a name or label. Chords form the basis of harmonic progressions that underpin much of Western music, defining its tonal structure and flow. Chord sequences and harmonic progressions provide a framework for understanding a piece’s key, tension, and resolution, making harmony an essential aspect for music analysis. Computationally, the automatic analysis of chord progressions has supported several tasks in information retrieval – from the detection of cadences [215], structures in music [60], to the introduction of harmonic similarity measures for cover song detection [227, 104], classification [305], and generation [67].

Available datasets containing chord annotations vary significantly across several dimensions. They differ in terms of data formats, with annotations stored in formats such as *LAB*, *CSV*, *TXT*, and *MXL*. The notational systems used for representing chords also vary widely, including *Harte*, *Leadsheet*, *Roman numerals*, and *ABC notation*. Finally, the datasets differ in the type of music representation they annotate, with some focusing on symbolic representation, where time is expressed in beats and measures, and others on audio data, where time is expressed in seconds. These variations underscore the need for harmonised data representations in MIR, highlighting the prominence and challenge of developing systems that are both syntactically and semantically interoperable in the context of harmony.

Taking harmony as a use case, this thesis investigates a workflow to harmonise symbolic data annotated in different formats and syntaxes, addressing both context- and content-based information about music. The goal is to explore whether it is possible to create a standardised workflow to unify existing datasets and to facilitate the creation of new corpora of harmonised data.

This leads to a key research question:

RQ2 *What strategies can be developed to create a large-scale, unified symbolic music dataset that standardises diverse digital formats and annotation practices, enhancing consistency and accessibility for music analysis?*

This pursuit of a harmonised symbolic dataset not only seeks to mitigate the fragmentation inherent in current music datasets but also aims to improve the accessibility and utility of harmonic data in music computing. The expected contributions of this work include the development of methodologies that can effectively unify diverse datasets and the demonstration of how such unified data can enable more sophisticated analyses and applications in a wide range of MIR tasks.

1.1.3 Exploring Harmonic Similarity: Leveraging Large-Scale Corpora for Deeper Musical Understanding

The creation of large, harmonised corpora of symbolic harmonic annotations opens new avenues for more comprehensive and scalable exploration of harmonic similarity at a scale that was previously unattainable. By unifying and standardising diverse datasets, we can now study harmonic similarity across a much broader range of music, encompassing various genres, historical periods, and styles. This unprecedented scale allows for more meaningful insights and deeper analysis of harmonic content, which were difficult to achieve with smaller, fragmented datasets.

Since the advent of MIR, similarity has served as a fundamental paradigm for organising musical datasets. Large music collections require effective organisation to support meaningful exploration, enabling users to uncover new connections and insights. As Pampalk notes, “the value of a large music collection is limited by how efficiently a user can explore it” [297]. While contextual metadata is crucial for structuring music collections, content-based similarity provides an objective framework for comparing pieces independently of metadata, which is often inconsistent or incomplete in music datasets.

Moreover, by comparing the harmonic structures of many pieces, it becomes possible to abstract rules and processes that define a particular musical style or genre. The analysis of these large-scale datasets can also improve our understanding of the generative processes behind music, revealing the evolutionary paths that

have shaped its development over time [394]. This deeper understanding not only contributes to the study of musicology but also has practical applications, such as generating new music based on learned patterns or refining algorithms for music classification and recommendation systems.

Most content-based music similarity research has focused on audio data [362, 124]. However, a significant drawback of these approaches lies in their reliance on end-to-end algorithms, which often fail to provide interpretable explanations for why certain tracks are considered similar. This lack of transparency can result in biased similarity measures and obscure the commonalities between distinct tracks, leading to challenges in understanding the reasoning behind the outcomes [231].

An alternative to these audio-based methods is symbolic similarity, which offers a more explainable and interpretable approach. Over the past decade, symbolic music similarity has been applied to a variety of tasks, including cover song detection [101], genre classification [12], variation recognition [157], music search [84], and plagiarism detection [409]. While melodic similarity has received substantial attention in this domain, the study of harmonic similarity has not garnered as much focus in recent years. To the best of our knowledge, the state-of-the-art methods in this area include the *Tonal Pitch Step Distance (TPSD)* [104] and the *Chord Sequence Alignment System (CSAS)* [175]. Moreover, available approaches to harmonic similarity tend to consider tracks as similar only when their global harmonic profiles align, offering limited insights into local harmonic similarities and hindering the possibility of exploring shared patterns among songs.

A further objective of this thesis is to develop a more nuanced and scalable approach to harmonic similarity. By leveraging symbolic harmonic annotations, the aim is to explore methods that account for both global and local harmonic structures. A key goal is to demonstrate that large corpora can support this type of research by enabling comprehensive, scalable studies of harmonic similarity that were previously not possible with fragmented data.

In this perspective, similarity measures, together with access to large corpora of harmonic data, open up new possibilities for creating exploratory tools that support musicological research and creative applications. Such tools enable researchers to navigate and analyse extensive harmonic datasets, revealing hidden patterns, connections, and trends across diverse musical works, genres, and historical periods. Additionally, these similarity measures may offer practical applications for composers, inspiring them with harmonic ideas drawn from a broad

range of genres and styles and sparking creativity through harmonic suggestions that may not be immediately apparent. This leads to the third research question:

RQ3 *How can novel harmonic similarity measures be developed to capture both global and local harmonic structures? How can these similarity measures be applied to large corpora to support scalable musicological research and assist creative applications?*

This research aims not only to establish advanced similarity measures that make full use of harmonised, large-scale corpora but also to demonstrate how these enriched datasets can drive innovation in both analytical and creative MIR applications. By exploring these avenues, we aim to validate the potential of integrated data as a foundational resource for improved musicological analysis and creative exploration.

1.1.4 Limitations of Symbolic Data Integration: A Multimodal Approach

Despite significant progress in symbolic data integration, large harmonic corpora still reveal limitations that hinder the full potential of data-driven approaches. Two key challenges persist: (i) the scarce diversity and balancing of the available harmonic datasets and (ii) the inherent ambiguity and subjectivity of chord annotations [302].

The first challenge refers to the lack of diversity and balance in the available datasets. For instance, the ChoCo dataset, although being the largest corpus of chord annotations to date in terms of size and diversity, is heavily skewed towards mainstream Western genres, with nearly 80% of the data derived from pop and rock music. This bias towards a few dominant genres is critical because chord vocabulary and harmonic structure can vary widely across musical styles, resulting in long-tail distributions that are notoriously challenging to model computationally [302], particularly in the context of Deep Learning (DL) applications. This issue, often referred to as *chord vocabulary imbalance*, restricts the system’s ability to accurately handle less frequent chords, further contributing to biased outcomes.

Another critical issue that cannot be addressed via data integration is the ambiguity and subjectivity inherent in chord annotations. Annotators may interpret harmonic structures differently, leading to significant variation in the labelling of chord sequences [227]. This problem arises because defining a chord in a musical

context can be highly subjective. For instance, distinguishing between a chord sequence and a melodic line is often open to interpretation. Additionally, annotators might vary in the level of detail they focus on, such as whether to include rapid approach chords or arpeggiated chords. They may also be influenced by the instrument playing the harmonic line, such as piano or guitar, rather than focusing on the broader harmonic structure of the piece. Furthermore, a specific set of notes can be labelled differently depending on the context and the harmonic function the annotator identifies within the piece.

A particularly famous case illustrating the subjectivity of chord annotation is the opening chord of The Beatles' *A Hard Day's Night*, which has generated decades of debate among musicians and analysts. Various interpretations have been offered, ranging from George Harrison's description of an F chord with a G on top to other music theorists proposing **G7sus4**, **G11sus4**, or even **Dmin11**.

These challenges highlight the need for a more comprehensive approach to analysing harmonic data, that goes beyond the integration of symbolic annotations alone. To address these issues, this thesis advocates for a multimodal approach, i.e. combining symbolic annotations and audio signals to capture multiple dimensions of musical content [66].

Multimodal integration offers several key advantages:

1. It enables the exploration of tasks that inherently require audio, such as *Audio Chord Estimation (ACE)*.
2. It mirrors the human approach to chord transcription by combining both *audio and symbolic data*, providing an effective means to address inter-annotator agreement.
3. Audio data provides additional information, such as *timbre, dynamics, and articulation*, that symbolic data alone cannot capture.
4. This approach aligns with the growing trend in the *MIR community* to integrate symbolic and audio data, offering a more comprehensive understanding of musical content [359, 66].

To create a multimodal dataset that integrates symbolic chord annotations with audio signals, the first step is to retrieve the corresponding audio and align it with the symbolic data. However, only about 12% of the 20,000 annotated tracks in the ChoCo dataset are aligned with audio, highlighting the urgent need for a more efficient method to align audio with chord annotations.

1.1. Problem Definition and Research Questions

An effective audio-to-chord alignment method would not only fill this gap but also enable the creation of new multimodal datasets. A promising approach is to leverage crowd-sourced chord annotations from platforms such as Ultimate Guitar¹, e-chords², and Chordie³, which collectively offer millions of annotated songs, particularly from underrepresented genres such as electronic, metal, hip hop, reggae, and country. These platforms present a valuable opportunity to expand MIR datasets beyond the mainstream Western genres.

However, these repositories typically lack timing and duration information, making them unsuitable for MIR tasks that rely on temporal alignment between audio and symbolic data.

Although various approaches in the literature focus on aligning audio with symbolic data—primarily using Dynamic Time Warping (DTW) algorithms [288]—none are specifically designed for aligning audio with chord annotations. Building on the case of harmonic annotations discussed in RQ2, this thesis proposes the development of a method for aligning symbolic chord annotations with audio, referred to as *audio-to-chord alignment*, and leveraging this approach to generate new multimodal data from crowd-sourced datasets.

We intend to address the following research question:

RQ4 *Is it possible to align chord annotations with audio without prior time information, thus enabling the creation of enriched, multimodal datasets?*

1.1.5 Limitations in Audio Chord Estimation: Chord Imbalance and Subjectivity

The proposed multimodal dataset, which integrates homogenised symbolic chord annotations from diverse sources and aligns them with audio signals, holds significant potential for advancing Audio Chord Estimation (ACE) – a critical task in MIR. ACE automates the transcription of chords directly from audio recordings, offering a scalable solution for music transcription and analysis. Its applications are far-reaching, impacting fields such as music analysis, musicology, content-based retrieval, and music education.

¹<https://www.ultimate-guitar.com/>

²<https://www.e-chords.com/>

³<https://www.chordie.com/>

Over the past two decades, research in ACE has made considerable progress, leading to notable improvements in the accuracy and efficiency of chord transcription [302]. However, despite these advancements, recent performance gains have stagnated, prompting some researchers to suggest that the task has reached a “glass ceiling”[56]. Notably, increasing the amount of training data and scaling computational resources have not resulted in significant improvements in ACE performance[302]. This plateau is largely due to ongoing challenges such as the aforementioned chord vocabulary imbalance and inter-annotator disagreement.

This thesis seeks to explore strategies for overcoming these obstacles, with a focus on improving our understanding of inter-annotator agreement, enhancing ACE performance, and better capturing harmonic representations from audio [23].

Starting with the evaluation of inter-annotator agreement in chord annotations [97, 227], current metrics often rely on binary comparisons, where a match between two labels is scored as one, and any mismatch is penalised with a score of zero. However, as noted by [266], treating all discrepancies with equal severity can result in unfair assessments. Binary evaluations often fail to account for shared harmonic features between chords that, while annotated differently, exhibit meaningful similarities. For example, a mismatch between a *G7* and a *Gsus4* may be treated as a complete error, despite both chords sharing significant harmonic tension and resolution characteristics.

To address these issues, we propose leveraging music theory to enhance the model’s understanding of chord annotations. By formalising the semantics of theoretical concepts such as consonance and dissonance and embedding them into our model, we aim to enable more context-aware interpretations of chord sequences. This theoretical framework will allow the model to better distinguish between similar chords and offer more nuanced interpretations of ambiguous or subjective harmonic structures.

Finally, we aim at evaluating the effectiveness of this approach in improving ACE and the resulting harmonic representation from the audio signal. By combining both symbolic and audio data, along with a deeper integration of music theory, we aim to overcome current limitations and establish a more robust framework for automatic chord transcription.

This leads to the final research question:

RQ5 *How can the integration of formalised music theory concepts into the training and evaluation of ACE models address chord vocabulary imbalance and inter-annotator agreement issues, improve overall performance in Audio Chord Estimation, and improve harmonic representations from audio?*

1.2 Thesis Contribution

This thesis addresses several key challenges in MIR and contributes to the fields of Knowledge Representation (KR) and ontology engineering, aligning its goals with the broader objectives of the *H2020 Polifonia Project*⁴, which funded this research. Polifonia seeks to preserve and reveal European musical heritage by using computational tools to extract and interlink knowledge from diverse sources. In this context, the thesis advances the project’s mission by developing multimodal datasets, algorithms for various MIR tasks, and formalised music-related ontologies, all directly tied to the Research Questions (RQs) introduced in the previous section. Moreover, the thesis also examines the limitations of current data integration approaches, especially in the context of harmonic data, and proposes and implements strategies to address these challenges.

These contributions target core issues in the field, such as data fragmentation, interoperability, and scalability, while also supporting Polifonia’s aims to enable memory institutions, researchers, and the public to explore the cultural and historical layers of musical artefacts.

The first major contribution responds to *RQ1*, which seeks to create a unified semantic model for representing both musical content and context. This contribution is realised through the development of the Polifonia Ontology Network (PON) [92, 88], a framework of ontologies that formalises the semantics of music representation, metadata, annotations, performance mediums, and historical sources. PON enables the creation of interoperable knowledge graphs from diverse music datasets, offering a solution to the fragmentation problem outlined earlier, both from a content- and context-based perspective. By extending the eXtreme Design (XD) methodology and incorporating a comprehensive set of 361 Competency Questions (CQs), released as the *PolifoniaCQ dataset*, the ontology design

⁴<https://polifonia-project.eu/>

is rigorously guided to ensure it captures the complexity and richness of musical knowledge. Moreover, a suite of Ontology Design Patterns (ODPs) specifically tailored to musical heritage further ensures that the ontologies effectively model different dimensions of music, providing a unified and interoperable framework for MIR.

The second key contribution addresses *RQ2*, which focuses on harmonising symbolic datasets to overcome the fragmentation of chord annotation datasets. This contribution is exemplified by the creation of *ChoCo: the Chord Corpus* [90], the largest existing dataset for musical harmony knowledge graphs. ChoCo integrates over 20,000 high-quality harmonic annotations from 18 heterogeneous chord datasets, achieving interoperability across various notation systems and metadata standards by leveraging the JAMS [203] data structure. The dataset harmonises symbolic and audio annotations by converting chord labels into three reference notational systems, with Harte notation [181] serving as the primary bridge. By standardising diverse chord annotations, ChoCo creates a unified dataset that enhances the study of harmonic structures and a workflow that enables the automatic generation of music knowledge graphs, thus supporting large-scale symbolic music analysis.

In response to *RQ3*, which investigates harmonic similarity, the thesis presents two methodologies: LHARP and Harmony. Both approaches leverage symbolic data to explore content-based similarity, addressing limitations in existing methods that focus mainly on global harmonic structures.

The large, unified harmonic corpus developed in response to *RQ2* (ChoCo) serves as a foundational resource for these methodologies. By standardising diverse chord annotations, ChoCo enables comprehensive harmonic comparisons across datasets, making it possible to apply both LHARP and Harmony for in-depth analysis of harmonic relationships.

LHARP [93] introduces a novel similarity function based on a variation of the Longest Common Subsequence (LCS) algorithm, tailored specifically for identifying local harmonic structures within chord progressions. This approach allows LHARP to capture nuanced, recurring harmonic patterns, which are organised into a graph-based exploration tool. Through this graph, users can visually trace harmonic relationships between pieces, enhancing both musicological analysis and content-based retrieval.

Harmony [91] explores a different approach, using a similarity function based

on Dynamic Time Warping (DTW) and an advanced segmentation algorithm developed specifically for this task. This combination allows the creation of a Knowledge Graph (KG) of harmonic patterns, where nodes are patterns linked by temporal and similarity-based relationships. Moreover, we demonstrate how this KG can support creative applications by enabling composers to explore harmonic ideas and variations, assisting in the composition process by suggesting musically plausible pathways based on established harmonic progressions.

In response to *RQ4*, which focuses on aligning symbolic and audio data, this thesis introduces ChordSync [316], a novel methodology for precise chord-to-audio alignment. ChordSync is designed to overcome the limitations of traditional alignment methods that often require pre-existing weak alignment between annotations and audio.

The method leverages the conformer architecture [170], a powerful neural network framework known for handling audio-based tasks with high temporal precision. By utilising this architecture, ChordSync enables the alignment of symbolic chord annotations to audio tracks at scale, providing the accurate timing and duration information necessary for robust analysis. Moreover, this approach opens up new possibilities for creating large, high-quality, audio-aligned chord datasets from widely available, crowd-sourced resources that often lack precise temporal alignment. In addition to supporting symbolic and audio integration, ChordSync facilitates essential MIR tasks like ACE, enhancing both research capabilities and practical applications.

Finally, the thesis addresses *RQ5* by tackling two significant challenges that limit symbolic harmonic analysis: *inter-annotator agreement* and *chord vocabulary imbalance*.

First, we perform a comprehensive analysis of inter-annotator agreement in chord annotations, utilising non-binary metrics to better capture nuanced harmonic similarities. Specifically, we extend the metrics proposed by McLeod et al. [266] by introducing perceptual consonance-based distance metrics. This extension demonstrates that inter-annotator agreement, when measured with these novel metrics, significantly improves, providing a more accurate reflection of the harmonic alignment between annotations.

Building on these findings, we propose an innovative ACE model that integrates consonance-based label smoothing [287] and focal loss mechanisms [246]. To further address chord vocabulary imbalance, we implement a chord decomposi-

tion approach inspired by McFee et al. [263], which decomposes chord predictions into separate classifications for chord root, bass, and note activations. This flexible approach allows the model to infer chord labels based on component predictions rather than relying on a fixed vocabulary, enhancing its capacity to represent diverse harmonic structures effectively.

Finally, we demonstrate how these implemented approaches improve the model’s ability to capture harmonic descriptors in audio, showing that the learned representations yield more robust and accurate harmonic annotation from audio data.

This thesis’ contributions can be summarised as follows:

- **Development of the Polifonia Ontology Network (PON) [RQ1]:** A modular, interoperable framework of ontologies to model diverse aspects of musical heritage, allowing for seamless integration of musical metadata, annotations, and KG. With PON we release a set of ODP and CQs to enhance modularity, reusability, and standardization in musical knowledge engineering.
- **Introduction of the Chord Corpus (ChoCo) [RQ2]:** A comprehensive dataset of over 20,000 harmonically annotated pieces, integrating 18 distinct chord collections using the JAMS standard and facilitating symbolic and audio alignment. ChoCo enables standardized chord annotation practices and data interoperability, making it a pivotal resource for large-scale harmonic analysis and KG construction.
- **Novel Harmonic Similarity Algorithms [RQ3]:** Two methodologies – *LHARP* and *Harmory*, which focus on local harmonic similarity. *LHARP* introduces a flexible similarity function for analysing patterns within harmonic sequences, while *Harmory* provides a cognitive model-inspired KG for creative applications and compositional support.
- **Development of ChordSync for Chord-to-Audio Alignment [RQ4]:** A novel conformer-based alignment method for synchronizing chord annotations with audio data, allowing integration of symbolic annotations into audio datasets. Furthermore, ChordSync facilitates multimodal dataset creation from crowdsourced sources, which often lack any timing reference.
- **Enhanced ACE Model with Consonance-Based Metrics [RQ5]:** An innovative ACE model that integrates consonance-based label smoothing,

focal loss mechanisms, and a decomposition-based approach to handle chord vocabulary imbalance and inter-annotator agreement. We demonstrate how this model better captures harmonic representations from the audio signal and outperforms state-of-the-art methods in transcribing chords from audio.

1.3 Thesis Structure

This thesis is organised into five main chapters, each addressing key aspects of music representation, dataset creation, ontology engineering, harmonic similarity, and multimodal approaches in MIR. Each chapter contains its own *related work* section, reflecting the diversity of topics covered.

The first chapter, *Background* (Chapter 2), introduces foundational concepts in music representation. Section 2.3 covers music metadata and content representations, examining both metadata (Section 2.3.1) and music content (Section 2.3.2). In Section 2.3.3, the focus shifts to *signal representations*, while the discussion of *symbolic representations* follows in Section 2.3.4, where symbolic data formats, such as MIDI and MusicXML, and their limitations are addressed. The chapter then explores *knowledge representation* of music in Section 2.3.5, highlighting the role of semantic models and ontologies to manage multirepresentational complexity. Finally, *multimodal approaches* are examined in Section 2.4, providing an overview relevant to symbolic and audio data integration.

Chapter 3 focuses on *ontology engineering* and the design of a unified semantic model for music data representation. Section 3.2 provides a review of existing ontologies and methods in ontology engineering. Section 3.3 outlines the adopted methodology, including requirement collection (Section 3.3.1) and the design of the PON in Section 3.3.2. The PON’s structure and interoperability capabilities are detailed in Section 3.4, with a focus on the Music Meta Module of PON in Section 3.5. Section 3.6 discusses the ontology’s application and reuse. The chapter concludes with a summary and future directions in Section 3.7.

Chapter 4 presents the development of ChoCo: the Chord Corpus. Related work is reviewed in Section 4.2, and the creation methodology of ChoCo is described in Section 4.3. Section 4.3.1 explains the dataset construction steps, with further details on the incorporated data, unified format conversion, and the Knowledge Graph generation. Descriptive statistics of ChoCo are provided in Section 4.3.2, with technical validation presented in Section 4.4. Section 4.5

Chapter 1. Introduction

discusses usage scenarios and applications, while data availability and licensing are covered in Section 4.6. The chapter concludes with a summary in Section 4.7.

Chapter 5, *Similarity*, explores harmonic similarity. Section 5.2 reviews the state of the art in symbolic harmonic similarity. Section 5.3 introduces LHARP, an algorithm for harmonic similarity exploration. Harmory is presented in Section 5.4, discussing its segmentation and similarity algorithms, Knowledge Graph construction, and implications for computational creativity. The chapter concludes with a summary in Section 5.5.

Chapter 6 discusses integrating symbolic and audio data. Section 6.2 reviews audio-to-score alignment and ACE methods. Section 6.3 presents ChordSync, a novel alignment technique, followed by an analysis of inter-annotator agreement in Section 6.4. Section 6.5 discusses ACE improvements using consonance-based label smoothing and focal loss mechanisms. The chapter closes with a summary in Section 6.6.

Finally, Chapter 7 summarises the contributions of this thesis by revisiting the research questions presented in Chapter 1, discussing the solutions developed for each question, and analysing perspectives for future work.

CHAPTER 2

Background

This chapter provides the theoretical and methodological foundation for understanding the core contributions of this thesis. By drawing on established concepts in music theory, MIR, and KR, it highlights the challenges of music representation and retrieval, particularly in the context of harmonic analysis and multimodal data integration.

The chapter begins by exploring fundamental concepts of *music theory* in Section 2.1, focusing on key concepts and definitions of music and their relevance to computational music analysis. In Section 2.1.2, the discussion covers the main musical dimensions (or facets [119]) such as melody, rhythm, and timbre. Harmony, a central focus of this thesis, is further detailed in Section 2.1.3, where the chords and harmonic structures are explored in greater detail.

Following this, the chapter turns to *Music Technology* in Section 2.2, where the field’s evolution and key tasks are discussed. This section connects theoretical musical knowledge with computational tasks, laying the groundwork for the subsequent analysis of data representation.

In Section 2.3, the chapter delves into the core topic of *music representa-*

tion, focusing both on the representation of metadata (Section 2.3.1) and music content (Section 2.3.2). The section then addresses the *signal representations* of music in Section 2.3.3, highlighting its strengths and the challenges it presents. This is followed by a discussion on *symbolic representation* in Section 2.3.4, which explores how symbolic music data like MIDI and MusicXML are used in computational tasks, and the complications introduced by fragmented notational systems. The final subsection, Section 2.3.5, focuses on *KR*, introducing semantic models and ontologies and their potential to address the multirepresentational challenges identified earlier.

Lastly, in Section 2.4, the chapter covers the emerging field of *multimodal approaches* in MIR, with Section 2.4.1 providing a definition of multimodality and challenges to integrating symbolic and audio data.

Through this exploration, the chapter establishes the necessary background for the thesis' contributions, especially in areas concerning music representation, harmonic analysis, and multimodal integration.

2.1 Music Theory and Structure

The purpose of this section is to provide the foundational music theory concepts that will serve as a reference throughout the thesis. These concepts are essential for the analysis and discussion of musical structures, particularly in the context of MIR.

Music transcends time and cultural boundaries, yet each historic epoch, culture, and subculture has created its own unique way of expressing itself musically. This wide variety of expression gives rise to what is referred to as the “Multicultural Challenge” [119]. Music theory endeavours to define and explain what music is by offering a generalized representation, yet the diversity of musical expressions necessitates a variety of theoretical frameworks. These theories differ based on the specific music they aim to describe, the cultural and geographical context of the theorists, and the purposes for which the theories are intended [177].

This chapter will delve into the music theories used to analyse the Western art-music tradition. While these theories provide profound insights into this particular musical repertoire, they offer limited perspectives on the broader, universal nature of music.

It is important to note that this section is not intended to be a comprehensive

reference for music theory. For readers seeking a more detailed and comprehensive account of the field, we refer to [235, 69, 177, 26]. The main aim of this section is to provide context for the topics discussed in the thesis, with a focus on the most explored aspects of music theory, such as harmony, while intentionally overlooking other topics and concepts that fall outside the scope of this research.

2.1.1 Fundamentals of Music Theory

In this chapter, we delve into theoretical concepts that are rooted in the common-practice period of Western music, which spans from the Baroque to the Romantic periods (ca. 1650. 1900). During these three centuries, compositions were often structured around a gravitational centre, a fundamental phenomenon known as *tonality*. This centre, or tonic, is typically a single pitch labelled using letters from *A* through *G*, with possible modifiers such as “flat,” “sharp,” “major,” and “minor.”

Such centres of gravity in music, providing a point around which all pitches orbit, have been a part of musical structures since antiquity and continue to resonate in contemporary music across various genres, including film scores, popular and commercial music, folk music, and jazz [235], albeit in evolving forms. While the strict adherence to tonal centres characterized much of earlier Western music, modern compositions often experiment with these foundations, incorporating atonal structures, polytonality, and microtonal music which challenge the traditional roles of the tonic [177]. Despite these innovations, the concept of a central pitch remains influential, adapted and reinterpreted in genres ranging from minimalism to progressive rock and modern jazz.

Pitch and Pitch Classes

The term *pitch* describes the attribute of a sound—such as an individual musical note—that determines its position on a scale. In Western culture, pitch is perceptually recognized as the attribute that allows sounds to be judged as “higher” and “lower,” in the sense traditionally associated with musical melodies [313].

Pitch is perceived based on what the ear interprets as the fundamental frequency of the sound, even in cases where this frequency is an auditory illusion, such as with difference tones, and not actually present in the sound wave [184].

The *frequency* of a sound is determined by the rate of vibrations produced by the sound source, such as the plucked string of a violin or the vibrating reed of



Figure 2.1: *Music score illustrating various enharmonic intervals, highlighting equivalent pitch pairs.*

a clarinet in response to airflow. These regular vibrations occur at a speed—often measured in Hertz (Hz)—which directly influences pitch: higher frequencies yield higher pitches and lower frequencies result in lower pitches. For example, a sound at 880Hz is typically perceived as higher than one at 40Hz .

When two pitches are related by a frequency ratio of $2 : 1$, they are separated by an octave. Despite the difference in frequency, pitches separated by an octave are perceptually similar, leading to the convention of labelling them with the same letter name (A through G), distinguished only by a number that defines the octave (e.g., A_3 , A_4) [235]. This cyclical nature of pitch perception is foundational in Western music theory, where the octave serves as a reference point for organizing pitches.

The concept of *pitch class* encompasses all pitches that share a specific relationship, such as being separated by one or more octaves. This relationship is termed an “equivalence” because, within a particular musical context, pitches within the same class are considered interchangeable or equivalent [339].

Pitches within the same pitch class bear the same letter name irrespective of their octave designation. For instance, the pitch class “C” includes all Cs across various octaves (C_1 , C_2 , C_3 , etc.). In Western tonal music, there are twelve distinct pitch classes within each octave, corresponding to the twelve semitones of the chromatic scale.

The concept of pitch class simplifies the manipulation and analysis of musical materials, especially in the study of harmonic progressions. In this context, the crucial aspect to consider is the relationships between pitches rather than the specific octaves in which they are played.

Pitches that sound the same but are named differently are termed *enharmonically equivalent*. For example, as depicted in Figure 2.1, $C\sharp$ and $D\flat$ are enharmonically equivalent, as are $A\flat$ and $G\sharp$. Although $C\sharp$ and $D\flat$ sound identical on a piano, they serve different functions within their respective musical contexts [69].

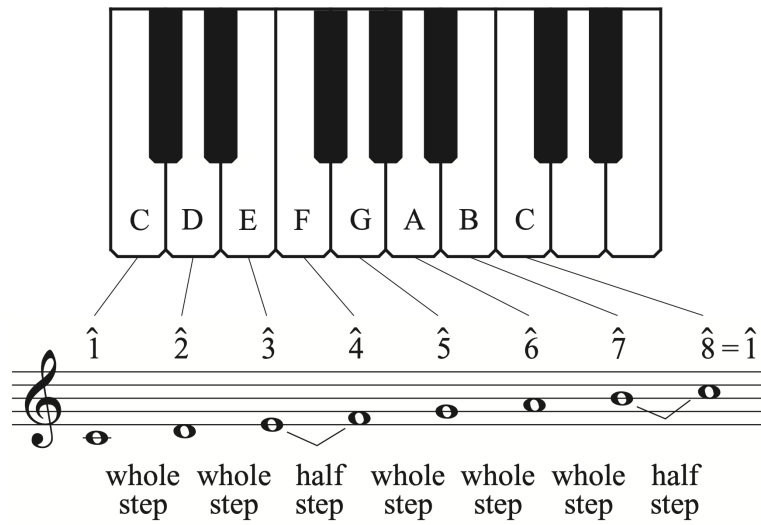


Figure 2.2: Visualization of the *C* major scale, displayed on both a piano keyboard and a musical staff, illustrating the corresponding notes across both representations and scale degrees.

Scales

A scale is defined as a sequence of pitches arranged in ascending or descending order [122]. The diatonic scale, predominant in Western music, is a seven-note sequence where each pitch, denoted by letters *A* through *G*, appears once per octave. The term “diatonic,” meaning “through the tones,” describes the division of the octave into seven steps, with the cycle completing by repeating the initial pitch at the octave [235].

The *major scale*, a primary example of a diatonic scale, adheres to a fixed pattern of whole steps (tones) and half steps (semitones), specifically arranged as $W - W - H - W - W - W - H$ (see Figure 2.2). This configuration imparts the major scale with its characteristic bright and uplifting sound. Conversely, *minor scales*, known for their more sombre tone, typically feature a lowered third scale degree, distinctly altering their emotional effect [49].

Each note in the seven-tone diatonic scale is assigned a functional name, reflecting its role within the scale. Figure 2.3 shows these names for each degree in both major and minor scales [26].

The pattern of intervals in the major scale can be transposed to any starting pitch, a process known as *transposition*. For instance, transposing the *C* major scale to start on *G* results in the *G* major scale.

Chapter 2. Background

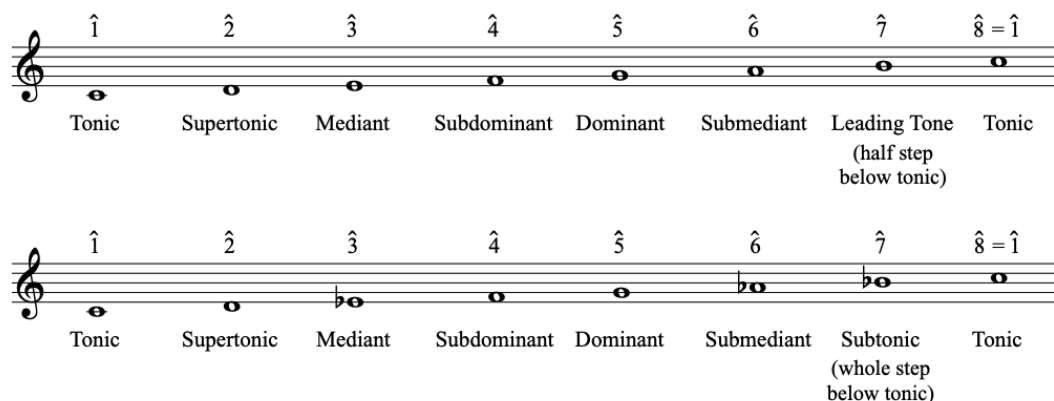


Figure 2.3: Scale degrees and their corresponding names within both major (*C major*) and minor (*C minor*) scales.

Major and minor scales dominate the landscape of Western classical tonal music. However, six other scales, known as modes or Church modes, were prevalent in Medieval and Renaissance music (pre-1600) and continue to appear in modern folk and popular music [369].

Additionally, several less common scales also exist in Western music. These include the *pentatonic scale*, and nondiatonic scales such as the *chromatic scale* and the *octatonic scale* [235].

Intervals

A fundamental concept in music theory is the interval, which refers to the distance between two pitches [247]. Intervals are designated by the number of diatonic notes (notes with distinct letter names) they span. For example, the distance from *C* to *F* is a fourth because it spans four letter names: *C*, *D*, *E*, and *F*. Intervals that encompass one octave or less are classified as simple, while those extending beyond an octave are termed compound.

Intervals possess qualities that further define their musical characteristics [235]. Intervals that include the tonic (keynote) and span to the fourth and fifth scale degrees of a major scale are termed *perfect*, as are unison and the octave, typically denoted with the letter *P* (e.g., *P4* for a perfect fourth). Intervals extending from the tonic to the second, third, sixth, and seventh degrees are classified as *major*, marked by an uppercase *M* (e.g., *M6* for a major sixth).

When a major interval is decreased by one half step, it becomes a *minor* interval, achievable by raising the lower note or lowering the upper note, and is

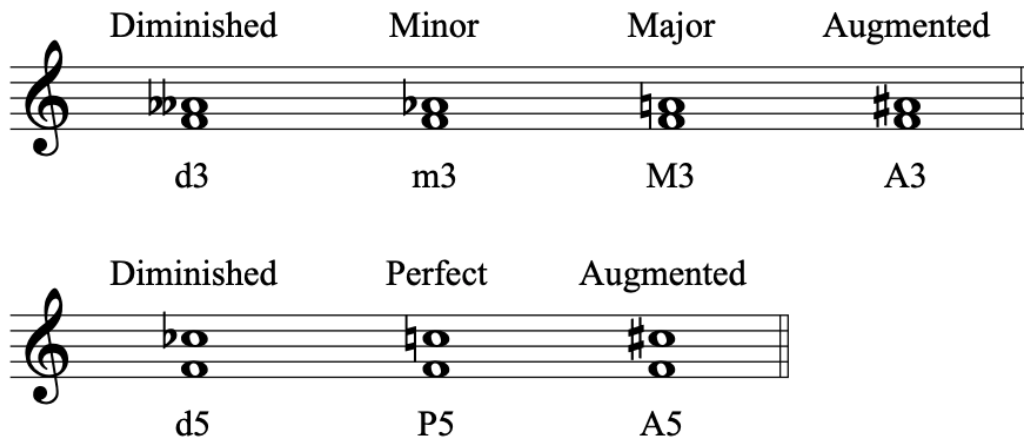


Figure 2.4: *Illustration of various intervals showing diminished, minor, major, augmented, and perfect qualities.*

denoted by a lowercase *m*. Conversely, augmenting a perfect or major interval by one half step transforms it into an *augmented* interval, marked by the letter *A*. Similarly, diminishing a perfect or minor interval by one half step changes it to a *diminished* interval, indicated by the letter *d*. Figure 2.4 displays examples of these interval types.

The *inversion* of an interval involves switching the positions of the tones, with the lower tone becoming the higher one, or vice versa [247].

Intervals can manifest in two forms: if the tones occur in succession, the interval is considered *melodic*. If they sound simultaneously, the interval is *harmonic*.

Harmonic intervals are further categorized as either *consonant* or *dissonant*. Consonant intervals, which sound harmonious and stable, are often utilized by composers at points of resolution. In contrast, dissonant intervals, characterized by tension and instability, are used to drive the music forward, demanding resolution and thus are unsuitable for concluding sections [369].

Tonality and Key

The concept of *tonality* refers to the orientation of melodies and harmonies towards a referential (or tonic) pitch class. In the broadest possible sense, however, it refers to systematic arrangements of pitch phenomena and relations between them [205].

Each key is built upon a *diatonic scale*. In any given key, the tonic is the central pitch around which other pitches revolve, creating varying levels of tension and resolution [235]. The structure of the diatonic scale organizes these pitches in a

Chapter 2. Background

pattern of whole and half steps, which gives each scale a unique tonal colour.

Within this scale, several pitches hold special significance in defining the tonality. Key members include the *dominant* (a perfect fifth above the tonic), which generates a strong sense of resolution when moving back to the tonic, and the *subdominant* (a perfect fourth below the tonic), which also plays a stabilizing role. The *mediant* sits midway between the tonic and the dominant, while other pitches such as the *leading tone* (just below the tonic) provide a strong pull back to the tonic, reinforcing the tonal centre [205].

The tonic is both the foundational pitch of the scale and the reference point for defining the key. For example, in C major, the note C serves as the tonic, while the surrounding pitches create a sense of stability and consonance around this central pitch. Modulations within a key can introduce variations in this tonal centre, as seen when different modes shift the scale's orientation [230].

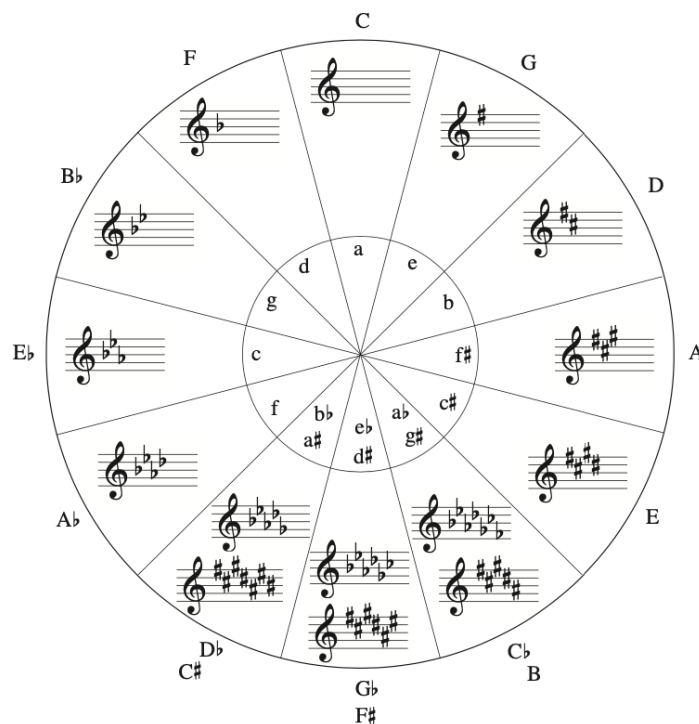


Figure 2.5: Illustration of relative major and minor keys using the circle of fifths, showing their interrelationships based on key signatures.

Key signatures are crucial for identifying the set of pitches used in a scale, typically denoted by sharps or flats that modify certain notes [69]. Each key signature represents both a major key and its relative minor. For example, a key signature with one sharp denotes G major and its relative minor, E minor.

Keys are systematically arranged within the *circle of fifths*, a structure that illustrates relationships among all major and minor keys based on their key signatures (see Figure 2.5). In this circle, adjacent keys like G major and D major are separated by a perfect fifth, indicating close tonal relationships. *Parallel keys* share the same tonic note but differ in mode (e.g., C major and C minor), while *relative keys* share the same key signature but have different tonic notes. The relative minor of a major key is determined by the sixth scale degree of that major key, which becomes the tonic of the relative minor. This systematic arrangement aids in understanding key relationships, providing a basis for composition, transposition, and music analysis [177].

Rhythm and Meter

In music, rhythm refers to the organization of sounds in time, distinguished by patterns of durations, beats, and accents. The fundamental unit of rhythm is the *pulse*, a series of undifferentiated and equally spaced clicks or taps. When these pulses are accentuated, they transform into a structured sequence of accented and unaccented *beats*, giving rise to what is understood as *meter*. Meter organizes these beats into recurring patterns within a measure, typically categorized as duple (strong-weak), triple (strong-weak-weak), or quadruple (very strong-weak-strong-weak) meters [26].

Beat division plays a crucial role in defining the structure of the meter. In simple meters, each beat is divided into two equal parts, while in compound meters, each beat is divided into three or more parts. Meter signatures, found at the beginning of a musical score, indicate the number of beats per measure and the rhythmic value assigned to each beat. For instance, in simple meters, the top number of the meter signature reflects the number of beats in a measure, while the bottom number denotes the type of note that represents one beat [235].

Asymmetrical meters, which do not conform to the regular division by twos or threes, feature irregular pulse groupings, such as $2 + 3 + 3 + 2$. These meters are common in the folk music of Eastern Europe and the Balkans, where they add a distinctive rhythmic character. Composers often use specific meter signatures to indicate how these complex rhythms should be interpreted, aiding performers in capturing the intended rhythmic feel.

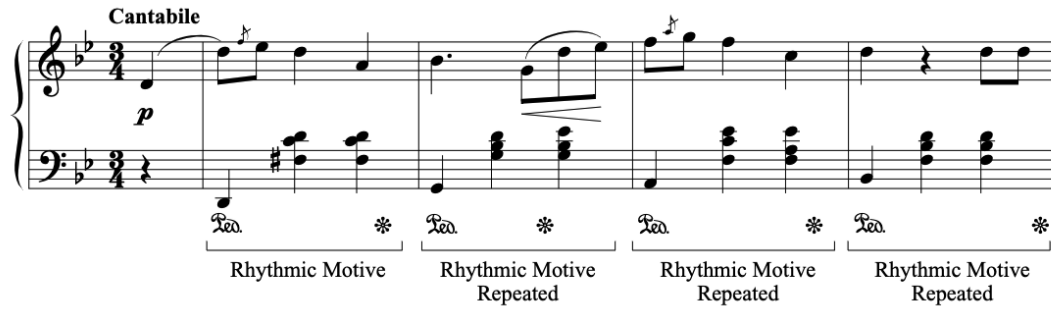


Figure 2.6: *Recurring rhythmic motive from Chopin’s Mazurka in G Minor, Op. 67, No. 2, illustrating the use of rhythmic patterns to create cohesion across the piece.*

2.1.2 Facets of Music

Building on the foundational theory concepts discussed earlier, music can be understood through multiple dimensions, or facets. Downie [119] identifies seven key facets of music: (i) pitch, (ii) temporal, (iii) harmonic, (iv) timbral, (v) editorial, (vi) textual, and (vii) bibliographic.

These facets highlight the multifaceted nature of music and its relevance across various fields.

However, it’s important to note that these facets are not mutually exclusive and often overlap in their application and interpretation. The complexity that arise from the complex interaction of the different facets has been labelled as the “multifaceted challenge.” For instance, the term “adagio” in a musical score might be relevant to both the temporal and editorial facets, depending on its use within the context of the piece [119]. Similarly, the harmonic facet is predominantly influenced by the interplay between the pitch and temporal dimensions, illustrating the interconnected nature of these musical aspects.

Pitch Facet

The pitch facet encompasses all aspects related to musical pitches, including scales, tonality, and key (c.f. Section 2.1.1). Within this domain, intervals are fundamental components, but they do not function in isolation; they contribute to broader musical constructs, notably *melody*.

Melodies are structured as sequences of pitches or intervals and are central to both the form and expression of music. They organize musical thought similarly to how written language uses sentences and paragraphs, grouping into coherent and meaningful units that enhance musical narrative and emotional depth [26].

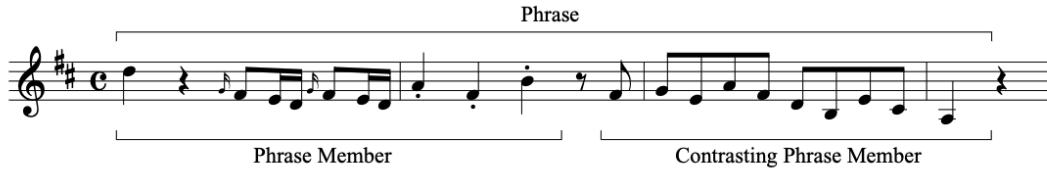


Figure 2.7: Illustration of phrasing in Mozart’s *Sonata in D Major, K. 284, I*, highlighting the structural and thematic development across measures 1-4.

A *motif*, or motive, is a brief, recurring figure that can unify a composition or a section within it. Motifs are crucial as they form the building blocks from which larger sections of music are developed. These can be melodic, involving repeated pitch patterns often paired with similar rhythmic patterns, or purely rhythmic, independent of melodic content [235].

Sequences further elaborate on motives by repeating them at different pitches, creating a chain of segments that enhance melodic development. This method was frequently employed in the eighteenth and nineteenth centuries to enrich compositions. In tonal music, sequences adhere to the diatonic scale, allowing for slight variations in transposition to maintain diatonic integrity.

Phrases represent complete musical thoughts and are typically defined by harmonic, melodic, and rhythmic cadences (see Figure 2.7). Often measuring around four measures in length, phrases can vary significantly in size. They may contain internal divisions known as phrase members, differentiated by breaks such as longer note values or rests. These members can be repetitive, sequential, or contrasting, contributing to the overall complexity of musical structure.

Temporal Facet

The temporal facet of music encompasses various elements that determine the duration and timing of musical events, including tempo indicators, meter, pitch duration, harmonic duration, and accents. These elements collectively form the rhythmic structure of a composition. Additionally, rests serve as indicators of silent durations within the music, providing breaks that contain no pitch information [119].

Temporal details in music can be expressed in absolute terms (such as a specific metronome marking), or using general descriptors (like *adagio* or *presto*). Rhythmic expressions can also include temporal distortions such as *rubato*, *accelerando*, and *rallentando*, which further complicate the rhythmic interpretation.

Moreover, different performance practices, especially in genres like Baroque and Jazz, often expect musicians to deviate from notated rhythms, adding another layer of complexity. These variations mean that a single rhythmic pattern can be represented in multiple ways, all leading to the same auditory outcome. Consequently, accurately representing and retrieving temporal information for music poses significant challenges.

Timbral Facet

Timbre, or tone colour, is often used as a “catch-all” term that encompasses all aspects of a sound’s quality except for pitch, loudness, and rhythm-related features. It also includes how changes in pitch, dynamics, and timing can alter the character of a sound [252]. Timbre is the key auditory attribute that enables listeners to distinguish between different musical instruments, such as a violin, an oboe, or a trumpet, even when they play notes at the same pitch and loudness [282].

Due to its elusive and complex nature, timbre is typically defined indirectly. It is the quality that allows listeners to perceive differences between sounds that are identical in pitch, loudness, and duration. This characteristic is crucial for distinguishing between instruments; for instance, it allows us to differentiate the sound of an oboe from a violin under identical pitch and volume conditions [284].

Moreover, performance techniques like *pizzicato* (plucking strings), muting, using the pedal, and various bowing methods also profoundly influence timbre. The application and notation of these techniques often blur the boundaries between timbral and editorial facets. While the choice of a performance method is an editorial decision, its audible impact pertains to the realm of timbre. In practice, effectively accessing and analysing timbral information typically requires audio or signal-based representations, which facilitate a nuanced recognition of tone colours through advanced signal processing techniques [119].

Editorial Facet

The editorial facet of music encompasses a wide range of performance instructions that significantly influence the interpretation and realization of a musical piece. This includes notations for fingerings, ornamentation, dynamics (e.g., *ppp* to *fff*), slurs, articulations, staccati, and bowings. Editorial information, however, presents various challenges due to its diverse forms; it can appear as iconic symbols (e.g., -, 3, !), textual descriptions (e.g., *crescendo*, *diminuendo*), or a combination

of both [119].

A significant issue within this facet is the absence of explicit editorial information, a common practice among composers prior to Beethoven and continuing with many composers thereafter. These composers typically assumed that performers would have the requisite knowledge and skill to interpret the music appropriately without detailed instructions [384]. This assumption can lead to discrepancies between different editions of the same work, complicating the selection of a “definitive” version for inclusion in a MIR system [120].

Textual Facet

The textual facet of music encompasses the written words associated with songs, arias, chorales, hymns, symphonies, and other forms, including the libretti of operas [191]. This facet highlights an important aspect of music information: the lyrics or texts can often be quite independent from the melodies and arrangements with which they are typically associated. This independence means that a specific lyric fragment might not always be sufficient to identify and retrieve a corresponding melody, and the reverse is also true [381].

Furthermore, many songs undergo translations into various languages, adding another layer of complexity to the textual facet. This multiplicity of textual settings for a single melody—or conversely, multiple musical settings for a single text—necessitates a careful approach when handling music information retrieval [221]. It is also crucial to acknowledge the vast amount of instrumental music that exists without any associated text, representing a significant portion of the musical corpus that relies solely on non-textual elements for its expression.

Bibliographic Facet

The bibliographic facet of music information includes details such as a work’s title, composer, arranger, editor, lyric author, publisher, edition, catalogue number, publication date, discography, and performers, among other aspects [119]. This facet is distinct in that it does not derive directly from the musical content itself but rather encompasses descriptive metadata about the musical work.

Like traditional bibliographic fields, the music bibliographic facet faces numerous challenges related to description and access. These challenges mirror those found in other domains of library and information science, where issues such as consistency and accuracy of metadata are ongoing concerns [296].

We further discuss the challenges that can be encountered when dealing with the bibliographic facet in Section 2.3.

Harmonic Facet

The harmonic facet of music, due to its significant relevance in the context of this thesis, warrants a dedicated section (c.f. Section 2.1.3).

2.1.3 Focus on Harmony

Harmony is a prominent dimension of the Western tonal music, also known as the “vertical dimension”, which is concerned with “*combining notes in music to produce a pleasing effect greater than the sum of its parts*” [74]. Harmony is a widely studied component in music theory [312, 351], and music analysis[188]; where functional harmony provides a set of rules for moving to and from the *tonic* – the most stable note in a piece, allowing to relate chords to each other, and to the main harmony. Moreover, harmony, along with metrical structure is ubiquitous: roughly speaking, every piece, in fact every moment of every piece, has a metrical structure and a harmonic structure [380].

The foundations of modern harmony began to take shape during the thirteenth century with the development of *organum*, progressing through the medieval period into the Renaissance. During the Renaissance (1450–1600), the concept of harmony evolved through the study and practice of *counterpoint*, which involved the interweaving of independent melodic lines [344]. Counterpoint emphasized the consonant and dissonant relationships between these lines, laying the groundwork for the vertical (harmonic) perspective of combining notes. This exploration of consonance and dissonance within multiple melodic layers became a cornerstone of harmony, as theorists sought to formalize the principles governing these relationships [70].

The Baroque period (1600–1750) saw the development of accompanying a melody with chords through the figured bass system [407], where keyboard performers improvised an accompaniment from a given bass line marked with symbols indicating the chords to be played. This practice was predominant throughout the Baroque era, not only in keyboard accompaniments but also in solo songs and ensemble compositions. The seminal work in the theory of harmony was Jean-Philippe Rameau’s “*Traité de l’harmonie*”, which introduced and discussed the inversion of chords, a concept that has significantly influenced theoretical

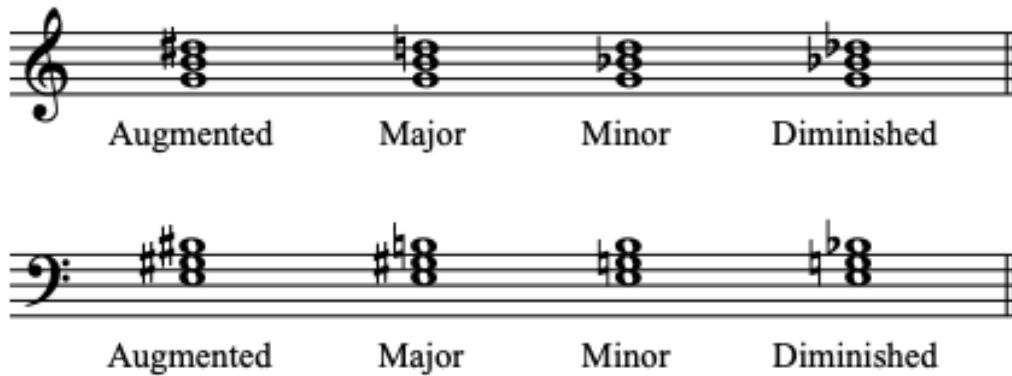


Figure 2.8: Examples of triads: augmented, major, minor, and diminished, illustrating the distinct interval structures that define each type.

perspectives in music [26].

Chords

Chords are the basic constituents of harmony, which jointly define the harmonic structure of a piece. Individually, a chord is defined as a “simultaneous occurrence of three or more music sounds, producing harmony” [70]. Depending on the notational system and the annotation conventions, a chord can be associated with a name, or label.

A *triad*, the simplest form of a chord, comprises any three-tone combination. It is identified by the root note upon which it is built—for example, a “C major triad” consists of the root *C*, along with a major third and a perfect fifth above it. Triads are classified into four types based on their quality: *major*, *minor*, *diminished*, and *augmented*. The arrangement of these intervals defines the overall sound and character of the triad [26]. Figure 2.8 shows examples of each of such triads.

The position of a triad within a piece, whether it appears in root position or as an inversion, significantly influences its harmonic function. In root position, the root note is the lowest pitch, while in inversions, either the third or fifth takes the lowest position, changing the chord’s texture and perception.

Further complexity is added by extending these triads to form larger chords such as *seventh*, *ninth*, *eleventh*, and *thirteenth* chords, each stacking intervals on top of the original triad [165], as shown in Figure 2.9.

The *seventh chord*, for instance, adds a note a third above the triad’s fifth,

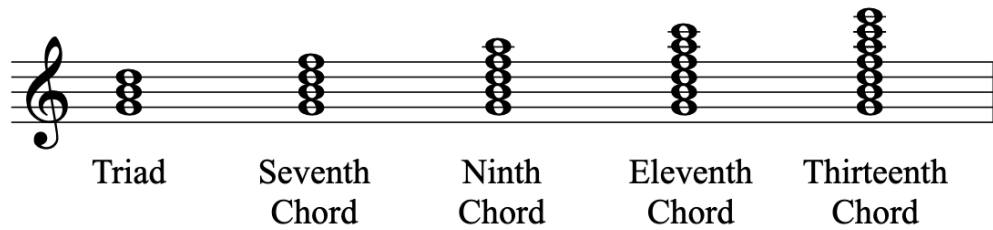


Figure 2.9: *Examples of chord extensions: Illustration of a major triad and its progressive extensions to seventh, ninth, eleventh, and thirteenth chords, built on the same root note.*

creating a distinctive interval of a seventh from the root in its root position. The dominant seventh chord, built on the fifth scale degree of the major and minor scales, combines a major triad with a minor seventh, contributing to its pivotal role in tonal harmony. Each type of seventh chord possesses a unique sound, dictated by the combination of the triad type and the seventh interval [70].

In musical analysis, Roman numerals are employed to identify and differentiate triads and their inversions based on their scale degrees. These analytical symbols help in understanding the placement and function of chords within the overall key structure. Additionally, figured bass notation, a system from the Baroque period, uses figures placed under a bass line to specify the intended harmony, streamlining the notation process for continuo players and emphasizing key harmonic elements [312].

Harmony not only structures the musical piece but also deeply influences its emotional impact. Major triads are often associated with positive emotions, while minor triads convey negative feelings. Diminished triads suggest tension and suspense, and augmented triads evoke a sense of unease or mystery. These emotional associations highlight the powerful affective role of harmony in music, resonating with listeners from diverse cultural backgrounds [225, 74].

Harmonic Progression

Harmonic progression refers to the sequence in which chords succeed each other within a musical composition. Historically, from the Baroque through the Classical and Romantic periods, composers have utilized harmonic progression as a primary organizational tool. The transition from one chord to another not only propels the music forward but also adds a dynamic element that is distinct from



Figure 2.10: *Illustration of a basic I – V – I harmonic progression, showing the movement from tonic to dominant and back to tonic, creating a sense of resolution.*

the contributions of melody or rhythm alone [177]. In tonal harmony, which pervades various musical styles, the structure of a piece is largely shaped by its chord progressions.

In tonal music, the tonic chord represents the pinnacle of stability. Chords that deviate from the tonic generally introduce tension, while those that return to the tonic tend to resolve this tension, providing a sense of fulfilment. Consequently, tonal compositions typically conclude with the tonic triad, and often commence with it as well.

The relationships between successive chords are governed by two principal forces, both related to the roots of the chords: (i) the relationship of the chords to the prevailing tonality, and (ii) the intervals formed by the roots of adjacent chords. Triads built on each scale degree relate back to the tonic triad, which serves as both the point of rest and the ultimate goal of harmonic progression. To fully understand harmonic movement, it is insightful to analyse chord progressions based on the interval between the roots of consecutive chords [26].

By examining the intervallic relationships between the roots of successive chords, harmonic progressions can be grouped into meaningful categories.

One of the most fundamental progressions in tonal harmony involves the dominant harmony leading back to the tonic, which establishes a sense of closure. The *I – V – I* progression, often written in this standard notation, begins on the tonic (providing a state of stability), moves to the dominant (which has a strong tendency to resolve back to the tonic), and finally returns to the tonic, offering a conclusive sense of arrival and completion [369].

Harmonic Cadences A *harmonic cadence* serves as musical punctuation, providing closure to a phrase or section of music. Cadences can vary in strength,

with some signalling the end of a complete musical thought (comparable to a period) and others suggesting continuation, akin to a comma [369]. Most cadences involve either the V or I chord, with the dominant chord frequently appearing as a seventh chord ($V7$).

Several types of cadences are commonly encountered in Western tonal music [26]:

- **Perfect Authentic Cadence (PAC):** This is the strongest cadence, consisting of a $V - I$ progression in major keys (or $V - i$ in minor keys), with both chords in root position and the tonic note as the highest sounding pitch in the final chord. The PAC provides a powerful sense of finality.
- **Imperfect Authentic Cadence (IAC):** This is a slightly weaker form of the PAC. It occurs when one or both chords are inverted, when the highest tone in the tonic chord is not the tonic itself, or when the diminished seventh chord ($vii^{\circ}6$) substitutes for the V chord.
- **Half Cadence (HC):** The half cadence ends on the dominant (V) chord, creating a sense of pause that anticipates continuation. Common forms include $I - V$, $IV - V$, and $ii - V$, with the Phrygian half cadence ($iv^6 - V$) as a notable variant in minor keys.
- **Plagal Cadence:** The plagal cadence moves from IV to I in major keys (or iv to i in minor keys). This cadence provides a softer sense of resolution than the authentic cadence.
- **Deceptive Cadence (DC):** In this cadence, the progression begins with V but resolves to a chord other than I , typically the vi chord in major keys (VI in minor). This progression provides an unexpected twist, delaying the sense of finality.
- **Rhythmic Cadence:** Phrase endings frequently feature characteristic rhythmic patterns that emphasize the cadence. These rhythmic cues add to the sense of closure or pause in the music.

These cadences, especially the V - I relationship in the PAC, are fundamental to the experience of resolution in Western tonal music. Cadences help to shape musical phrases, each resembling a self-contained musical thought, with clear beginnings and endings.

2.2 Music Technology

2.2.1 Historical Background

The convergence of music and computational sciences has significantly reshaped the ways we study, interpret, and engage with music. In recent decades, progress in computational methods has provided researchers with powerful tools to analyse extensive music collections and uncover structural aspects of music with unprecedented depth. These advancements have enabled the examination of vast quantities of musical data—surpassing traditional human limitations—and have transformed fields such as musicology, music analysis, and creative music applications.

This transformation began in the mid-20th century when the first computational applications to music were conceptualised in the 1950s and gradually implemented during the 1960s [30]. At that time, computational models enabled researchers to address challenges in music analysis that traditional methods could not manage—such as identifying patterns across large datasets, conducting statistical analyses, and developing formalised representations of music. By the late 20th century, the digitisation of music and the rapid growth of digital archives accelerated the demand for intelligent systems capable of managing and retrieving musical information. This development laid the foundation for MIR, a field dedicated to creating models for handling and analysing vast digital music collections.

MIR has developed as a distinct field of research from the 1990s onwards, culminating in the establishment of the International Society for Music Information Retrieval (ISMIR) ¹, a dedicated conference series at the turn of the millennium. The field addresses unique challenges in music retrieval, adopting since its foundation an interdisciplinary approach:

MIR is a [...] interdisciplinary research area encompassing computer science and information retrieval, musicology and music theory, audio engineering and digital signal processing, cognitive science, library science, publishing, and law. Its agenda, roughly, is to develop ways of managing collections of musical material for preservation, access, research, and other uses. [119]

Its goal is to manage and utilize musical materials for various purposes like preservation, access, and research. Content analysis, which involves the automatic

¹<https://ismir.net/>

extraction of music descriptors directly from audio, along with the development of innovative interfaces and infrastructure, are central to MIR’s objectives [221].

2.2.2 Music Information Retrieval Tasks

MIR integrates various paradigms for accessing music information, tailored to suit different user interactions and needs, as analysed by Knees & Schedl [221]. These paradigms include:

- **Retrieval:** This paradigm involves users actively expressing a specific music-related need through a query, which can be in forms such as text, symbolic music representation, or audio. The system retrieves and potentially ranks results—audio clips, scores, or metadata—corresponding to the query. This method aligns with traditional information retrieval but is specialized to accommodate the distinctive characteristics of music data.
- **Browsing:** Unlike retrieval, browsing allows users to explore a music collection without a predetermined goal. This paradigm supports an interactive and iterative process facilitated by intuitive user interfaces, enabling users to discover music items serendipitously.
- **Recommendation:** In this paradigm, the system proactively filters and suggests music items based on the user’s past actions or stated preferences, which may be explicitly provided or inferred through behavioural patterns such as previous queries or playback history. Recommendations provide a personalized experience by predicting user preferences and offering music choices accordingly.

These paradigms are applied in various contexts within the field of MIR, ranging from systems that allow queries by humming to sophisticated algorithms for theme detection using symbolic data representations like Musical Instrument Digital Interface (MIDI) [282]. Further, technologies such as music fingerprinting identify songs from brief audio clips, even in noisy environments, and cover song identification algorithms analyse components like melody and harmonic progressions to find different renditions of the same song [221].

While a single, unified taxonomy for MIR tasks is challenging and beyond the scope of this work, various resources provide substantial insights and methodologies for specific tasks. Textbooks such as [282, 241] and ISMIR conference

proceedings are instrumental for gaining a deeper understanding of these tasks. Furthermore, [252] proposes a semi-comprehensive taxonomy of MIR tasks, categorized broadly as follows:

- **Tonality and Harmony:** Mode, chord, and key detection.
- **Melody and Pitch:** Melody estimation, pitch and multi-pitch detection, note tracking, automatic music transcription.
- **Rhythm:** Onset detection, beat and downbeat tracking, metre estimation, tempo estimation.
- **Temporal Alignment:** Score following, audio-to-score alignment, score alignment.
- **Source Separation:** Musical instrument source separation, harmonic-percussive source separation.
- **Timbre-related Tasks:** Musical instrument identification, playing technique detection.
- **Clip-level Classification:** Music tagging, genre recognition, emotion/mood recognition.
- **Content-based Audio Retrieval:** Audio identification, audio matching, cover song detection.
- **Temporal Segmentation:** Music detection, music structure segmentation, time boundary identification.
- **Visual Score Input:** Optical music recognition and subtasks, including staff line identification and music symbol identification.
- **Performance-related Understanding:** Technique identification, performer identification, performance assessment, difficulty estimation.

While a comprehensive overview of all efforts and tasks within the MIR field is unfeasible, Ma et al. [252] highlight several methodological trends. They observe that the bulk of MIR research primarily utilizes audio as the input modality, with a smaller proportion relying on symbolic representations. The focus is predominantly on Western tonal music, relegating non-Western cultures and folk or traditional music to lesser prominence. A key differentiator among these tasks lies

in the required temporal granularity, ranging from clip-level classifications such as audio tagging to tasks demanding fine temporal resolution, such as pitch detection and onset detection. This variability in temporal demands, coupled with the global diversity of musical cultures, presents challenges in developing a universal music foundation model. Nevertheless, Ma et al. also note that the intrinsic connections among various tasks, such as the relationship between onset detection and beat tracking, could lead to the development of versatile music representations and MIR models capable of addressing multiple tasks, albeit potentially at the cost of reduced musical diversity.

2.2.3 Harmony in MIR

Harmony plays a pivotal role in MIR as it forms the foundation of numerous analytical tasks that interpret the vertical dimension of music. Key harmony-related MIR tasks include:

- **Audio Chord Estimation (ACE):** ACE involves the automatic detection of chords in audio recordings, returning a symbolic chord sequence that represents the harmonic structure. It is one of the fundamental tasks in MIR, with applications in music transcription, similarity, and analysis [259, 146, 411, 267, 412, 343].
- **Key Detection:** This task identifies the key of a musical piece, which provides the tonal context for interpreting harmonic progressions and melody [424, 59, 220, 133].
- **Cadence Detection:** Cadence detection identifies harmonic cadences, which mark the ends of musical phrases and are critical for understanding musical form and structure [33, 215].
- **Harmonic Similarity:** This task measures the similarity between harmonic progressions, facilitating tasks such as cover song identification and music recommendation [104, 3, 100, 175].
- **Functional Harmonic Analysis:** Functional harmonic analysis seeks to annotate chords with their functional roles (e.g., tonic, dominant), providing a deeper layer of harmonic meaning and enhancing the understanding of chord progressions [289, 61, 76, 331].

- **Harmonic Change Detection:** This task detects points in a piece where significant harmonic changes occur, which can reveal shifts in mood, tension, or form within the music [329, 108].
- **Harmonic Predictive Modelling:** In predictive modelling, systems are trained to anticipate upcoming harmonic structures, often used in generative applications or music prediction algorithms [82, 403].

These harmony-related MIR tasks are not solely ends in themselves; they also play a critical role in enhancing broader MIR paradigms, such as retrieval, recommendation, and browsing, by enabling more sophisticated and contextually rich interactions with music data. Moreover, harmony analysis contributes to improving other MIR tasks, including version identification [345], genre classification [201], and music structure segmentation [60], enriching the scope and depth of computational music analysis.

2.3 Music Representation

A foundational aspect of MIR research is the representation and structured organisation of music, which serves as the basis for different methods of accessing and interacting with musical content. Historically, the paradigms in MIR have revolved around two primary forms of music information access: *context-based MIR* and *content-based MIR* [221].

- **Content-Based MIR** focuses on the direct analysis of the audio signal itself, covering elements that can be extracted such as rhythm, timbre, melody, harmony, and even the mood of the music piece. These aspects represent the core components of a music piece that contribute to its unique auditory signature and perceptual impact.
- **Context-Based MIR** involves aspects of music that cannot be directly derived from the audio signal. This includes metadata such as reviews, liner notes, album artwork, country of origin, recording decade, and marketing strategies. These elements, often termed as cultural features, community metadata, or context-based features, provide a broader cultural and contextual perspective on the music, influencing how it is perceived, understood, and valued across different listener communities.

This section explores these two dimensions in depth. Section 2.3.1 examines the unique characteristics and challenges of representing musical metadata, while Section 2.3.2 focuses on approaches to capturing and representing musical content.

2.3.1 Representing Musical Metadata

In today’s increasingly digitized society, the wealth of knowledge surrounding music—commonly referred to as metadata—has grown exponentially, becoming a crucial resource for the music industry and beyond. Given the distinct cultural and perceptual aspects of music, designing suitable representations for this metadata demands careful and specialised approaches.

The representation of musical metadata involves creating structured descriptions that enhance the understanding, management, and accessibility of music within digital systems. This approach aligns with the “bibliographic facet” of music, which emphasises descriptive information about music rather than its direct sonic or notational representation [119].

To make this vast body of musical knowledge easily accessible to listeners and efficiently manageable by machines, it is essential to develop systems capable of interpreting and utilising music metadata effectively. As François Pachet [296] notes, the task of music knowledge management revolves around two key objectives: (i) constructing meaningful and maintainable descriptions of music, and (ii) leveraging these descriptions to build robust access systems that allow users to navigate large music collections with ease.

Musical metadata can be classified into three primary categories: editorial, cultural, and acoustic. Each category plays a distinct role in how music is described, organised, and accessed within digital ecosystems, and together they form the backbone of modern music information retrieval systems [296].

Editorial Metadata

Editorial metadata refers to information that is manually curated by experts, such as details about albums, songs, artists, recording dates, and composers. This type of metadata is provided through deliberate editorial processes, often by authoritative sources. It can range from straightforward administrative information to more subjective data, such as artist biographies or genre classifications. The challenge with editorial metadata lies in ensuring consensus on subjective content (e.g., musical genre taxonomy) and keeping the metadata up-to-date as new music, artists,

and genres emerge.

Editorial metadata can also be collaboratively produced by users, as seen in databases like MusicBrainz². Collaborative efforts raise different management challenges compared to prescriptive, expert-driven systems, yet they offer scalable solutions for growing music collections.

Cultural Metadata

Cultural metadata emerges from broader social and cultural contexts rather than being directly input by individuals. It is derived from patterns and associations observed in large datasets, such as user behaviour, playlists, web searches, and text sources. Methods like collaborative filtering or co-occurrence analysis extract relationships between musical items (e.g., artists, songs) based on their proximity in cultural contexts (e.g., search engine results or playlists).

Cultural metadata can provide insights into the similarity between artists or the association of words (like genres) with specific artists. This approach helps in generating automatic music recommendations or genre classifications, although cultural metadata may differ from expert-defined editorial metadata.

Acoustic Metadata

Acoustic metadata is generated through direct analysis of the audio signal itself, independent of any external or manually provided information. It aims to capture objective musical content such as tempo, rhythm structure, or instrument types. Techniques for extracting acoustic metadata have made significant progress, especially in rhythmic and beat detection, but challenges remain in developing robust extractors for more complex musical features such as mood, energy, or melody.

Some acoustic descriptors apply to an entire track (e.g., tempo), while others, like melodic contour, depend on specific positions within the track. Advanced techniques can even infer the structure of a piece of music, leading to applications like automatic music summarization. Standards like MPEG-7 [229] aim to standardize the representation of these descriptors, but the reliability of extractors is still a limiting factor.

²<https://musicbrainz.org/>

2.3.2 Representing Musical Content

One of the most significant challenges in the computational study of music is its representation. This complexity reflects music’s multifaceted nature, as previously outlined. The core difficulty lies in integrating the empirically measurable aspects of music with its non-empirical, interpretive dimensions, as Roger Dannenberg highlights:

If musical information was well-understood and fixed, then music representation would be a much simpler problem. In reality, we do not know all there is to know, and the game is constantly changing. [83]

Historically, the study of music was facilitated by the advent of music notation. Initially, scores were introduced to record music, enabling musicians to replicate a piece and thus preserve it across performances [25]. However, notation brought a host of issues, many of which remain unresolved, as scores symbolise rather than represent music directly. This symbolic representation has notable limitations:

People don’t play musical rhythms as written, often they don’t play the pitches as written, and that’s not because they play it wrong but because the notation is only an approximation. And that’s before we start thinking about all those dimensions like timbre and texture that aren’t directly represented in the notation at all. All of these missing elements have to be supplied by the performer or the musicologist if you’re to make sense of the score as music. In the absence of that there’s a real sense in which you’re studying scores and not music, and there’s also a real sense in which that’s what traditional musicology was set up to do. [74]

In the 19th century, audio recordings became possible, providing a more detailed and systematic form of music storage than notation. This development underscored a dichotomy between the symbolic representation level embodied by printed music and the tangible, non-symbolic level of audio signals [83].

This dichotomy deepened with the arrival of computers. Representing musical content in a machine-readable format raises numerous questions, addressing both the digitalisation of audio signals and the structuring of music’s essential features.

Representation Levels of Music

Analysing music representation means dealing with a long series of issues scholars have dealt with for decades. Pieces of music are not physical objects, so they have no single “ground” manifestation. The question about how to represent a piece of music to be processed by a computer, therefore, does not have a single answer [255].

Music information can be represented in technical systems through various types, each designed to capture different aspects of musical content depending on the needs of the system. Hugues Vinet [396] defined four key categories of music representations:

1. Physical representations
2. Signal representations
3. Symbolic representations
4. Knowledge representations

These categories span from concrete audio data to abstract knowledge structures, providing a framework for understanding how music is processed, analysed, and interacted with in computational environments.

The first and most fundamental type of music representation is the *signal representation*, which captures the raw audio signal either through recording or electronic synthesis. This representation, as an amplitude function of time, does not inherently understand the musical content; it is purely a continuous flow of data that could represent any kind of sound, musical or not. Despite various coding methods, signal representations—whether analogue or digital—focus on transmitting the auditory signal without direct reference to musical structures, making them suitable for tasks requiring precise sound transmission and manipulation.

In contrast, *symbolic representations* are inherently content-aware, encoding discrete musical events such as notes, chords, and rhythms. These representations are formalised according to concepts from music theory, allowing systems to process and interpret musical content at a higher level of abstraction. Symbolic representations account for discrete time events and event states, making them appropriate for applications like music notation, transcription, and composition. However, they do not capture the full complexity of audio signals, such as timbral nuances or continuous dynamic changes.

Beyond symbolic and signal representations, *physical representations* capture the spatial and physical properties of sound sources and scenes. These representations are particularly useful for applications involving acoustic models, 3D audio simulations, and virtual or augmented reality systems. They account for sound source properties like directivity patterns and radiation, as well as spatial characteristics of sound environments, providing crucial data for synthesising spatial audio experiences.

Finally, *knowledge representations* are designed to encode structured formalizations of musical knowledge for specific applications, such as digital music libraries or music information retrieval systems. Unlike the other levels, knowledge representations rely on language structures and qualitative descriptions to capture high-level characteristics of musical works, such as genre, instrumentation, or performance qualities. These representations are abstract and do not necessarily have musical specificity, but they are crucial for describing global attributes that cannot be easily derived from signal or symbolic data alone.

However, Vinet highlights that, typically, different representation types are self-contained, with limited interoperability. Standard musical applications—such as sequencers, score editors, audio processing modules, and synthesizers—usually manage signal and symbolic representations either separately or with minimal interaction [396].

Furthermore, it is important to outline the desired characteristics of a symbolic music representation system. Wiggins et al. [404] propose to evaluate these representation systems according to two dimensions: *expressive completeness* and *structural generality*. Expressive completeness refers to the ability of the system to recreate the original content from the represented data [75]. Structural generality relates to the ability to represent and manipulate a wide range of high-level structures. For instance, while raw audio is highly expressive, containing rich performance details, its structural generality is limited as extracting structured information such as tempo or chords is challenging. Conversely, MIDI formats, though less expressive particularly in capturing timbral details, offer greater structural generality, facilitating the manipulation of structured musical data.

In this Section we are going to explore audio, symbolic and signal representations, providing details and examples.

2.3.3 Audio Representation of Music

The most information-rich representation of music comes from analogue signals, which capture sound in its rawest form. Digital audio formats, however, are used to convert these signals into a format that computers can process, with each format ranked by the information quantity it holds per time unit. This quantity depends on factors such as sampling rate and bit depth, which together determine the resolution and entropy of the digital audio signal [396]. While raw sound waveforms offer maximal expressive detail, they lack structural information, making it difficult to derive meaningful insights without further processing.

Sound originates from the vibrations of an object, which cause air molecules to oscillate, creating waves of alternating pressure. These waves travel through the air and can be detected by a listener or captured by a microphone [282]. To be processed by a computer, recorded sound must be digitised, transforming it from a continuous to a discrete representation. This digitisation occurs through sampling, where the amplitude of the sound wave is measured at equidistant time intervals, creating a discrete-time signal from the continuous waveform.

Despite their high fidelity, waveforms do not inherently reveal frequency information, which is crucial for many MIR tasks. Various transformations and algorithms have been developed to extract structured information from these signals, each with unique advantages and limitations. In the following section, we outline four commonly used audio representations in the MIR field, each designed to enhance specific aspects of musical analysis.

Log Mel Spectrogram is a pivotal audio representation technique that merges signal processing with psychoacoustic principles to closely mimic human auditory perception [282] (c.f. Figure 2.11a). This process begins by converting audio signals from the time domain to the frequency domain using the Fast Fourier Transform (FFT). It then utilizes the Short-Time Fourier Transform (STFT) to analyse these signals over time, segmenting the audio into overlapping frames and applying FFT to capture temporal dynamics. Mel band-pass filters are applied to the STFT outputs to transition to the Mel scale. These filters accommodate the non-linear nature of human pitch perception by grouping frequencies into bins that are linearly spaced at lower frequencies and logarithmically at higher frequencies [282]. The log Mel spectrogram is extensively used in music generation tasks due to its perceptual relevance [183, 142].

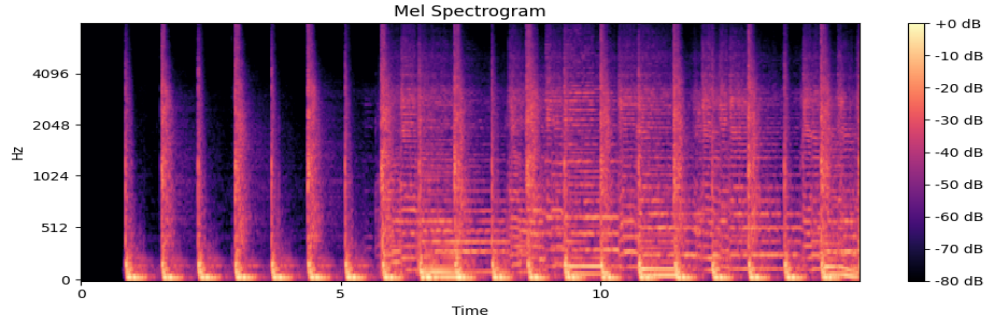
Mel-Frequency Cepstral Coefficients (MFCCs) are a critical audio feature extraction technique that encapsulates the characteristics of human speech perception [65] (c.f. Figure 2.11b). Similarly to Log Mel Spectrograms, the process initiates with the conversion of audio signals to the frequency domain via FFT, followed by the application of STFT for analysing temporal changes. Next, Mel filters are employed to emulate the non-linearity of human ear. The computation of MFCCs involves taking the logarithm of the energies in each Mel filter, followed by a Discrete Cosine Transform (DCT). Owing to their effectiveness in capturing the configurations of the vocal tract, MFCCs are predominantly used in speech recognition and speaker identification applications [386, 253].

Constant-Q Transform (CQT) enhances audio processing in music analysis by providing a log-frequency spectrogram aligned with the musical scale [1] (c.f. Figure 2.11c). Unlike the linear Fourier Transform, the CQT operates on a logarithmic scale that reflects the exponential nature of musical pitch, facilitating the extraction of note frequencies. A distinctive feature of CQT is that the ratio of centre frequency to bandwidth remains constant (denoted by Q), which allows variable filter lengths that optimize performance. While not as popular as the log Mel spectrogram, the CQT is beneficial for tasks such as music Representation Learning (RL) [244], particularly when related to harmony [301, 211].

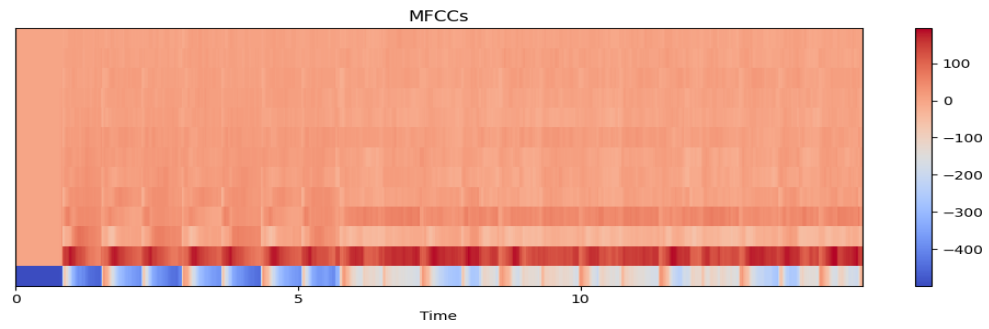
Furthermore, to better capture harmonic relationships in music audio signal, the Harmonic Constant-Q Transform (HCQT) was introduced. The HCQT is a three-dimensional array, indexed by harmonic, frequency, and time: $H[h, t, f]$, where it measures the h -th harmonic of frequency f at time t . The index $h = 1$ represents the fundamental harmonic, and $H[h]$ denotes the h -th harmonic of the base CQT $H[1]$.

Chroma Features or *chromagrams*, capture the twelve pitch classes of Western music, distilling the harmonic essence of a composition regardless of timbre or instrumentation (see Figure 2.11d). As described in Section 2.1, human perception of pitch is periodic; pitches differing by an octave are perceived as similar in “color” and play a related harmonic role. This perceptual property allows each pitch to be divided into two components: *tone height* (the octave number) and *chroma* (the pitch spelling attribute within the set $\{C, C\#, D, \dots, B\}$). Enumerating chroma values, this set can be mapped to $[0 : 11]$, where 0 corresponds to chroma C , 1 to $C\#$, and so forth. A pitch class, then, is the set of all pitches sharing the same chroma, such as $\{\dots, C_0, C_1, C_2, C_3, \dots\}$, and for simplicity, the terms

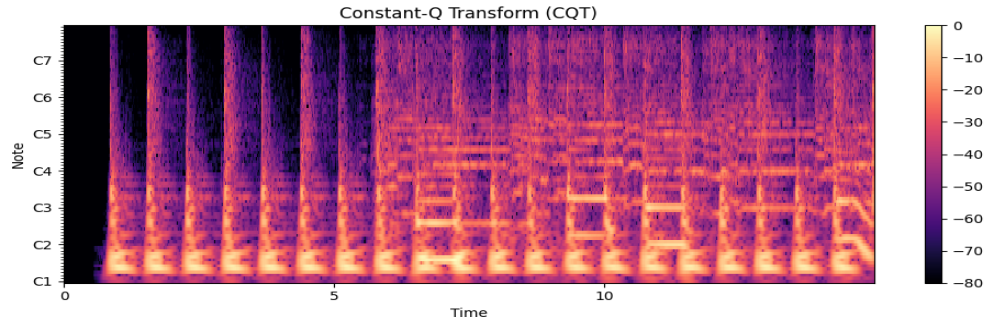
2.3. Music Representation



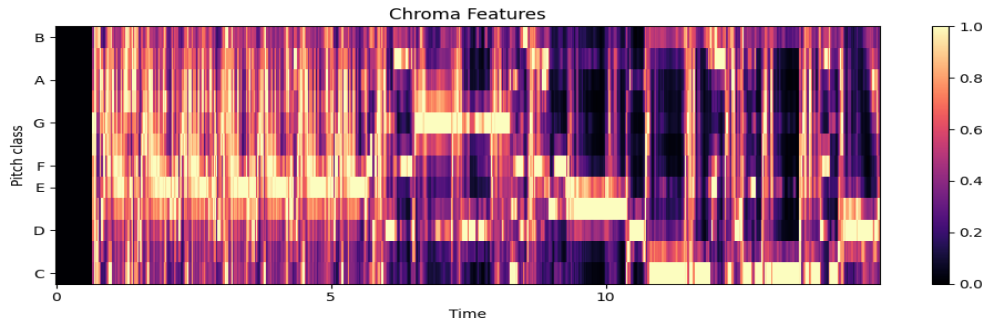
(a) *Mel Spectrogram*



(b) *MFCCs*



(c) *Constant-Q Transform (CQT)*



(d) *Chroma Features*

Figure 2.11: Audio features extracted from the first 15 seconds of “Do I Wanna Know?” by Arctic Monkeys. (a) shows the Mel spectrogram; (b) depicts the MFCCs; (c) presents the Constant-Q Transform (CQT); and (d) displays the Chroma features.

chroma and pitch class are often used interchangeably.

Chroma features aggregate spectral information associated with a pitch class into a single coefficient. Formally, given a pitch-based log-frequency spectrogram $Y_{\text{LF}} : \mathbb{Z} \times [0 : 127] \rightarrow \mathbb{R}_{\geq 0}$ as defined in equation (3.4), a chromagram $Z \times [0 : 11] \rightarrow \mathbb{R}_{\geq 0}$ can be derived by summing all pitch coefficients that share the same chroma:

$$C(n, c) := \sum_{\{p \in [0:127] : p \bmod 12 = c\}} Y_{\text{LF}}(n, p) \quad (2.1)$$

where $c \in [0 : 11]$ represents each chroma class.

Because of their ability to distil music into pitch class profiles, chroma features are effective in identifying key harmonic elements like chords, key signatures, and modulations. Their alignment with the equal-tempered scale of Western music makes them particularly valuable for applications in melodic transcription [22] and chord recognition [146]. However, chromagrams can exhibit noise in lower-frequency regions, and harmonic overtones often spread energy across multiple chroma bands. For instance, when playing C_3 , the third harmonic resonates in the G_4 chroma band, and the fifth harmonic in E_5 [282].

2.3.4 Symbolic Representation of Music

Symbolic music representation encodes musical elements into structured, computer-readable formats, allowing detailed manipulation and analysis of musical data [255]. Unlike audio recordings, symbolic representations capture abstract musical aspects—such as pitch, rhythm, harmony, and structure—independently of any specific performance, making them versatile tools for analysis and retrieval.

To be effective, Music Representation Systems (MRSs) must meet several essential requirements based on the nature of musical data. First, they need to be *multi-dimensional*, capturing both quantifiable elements like pitch and tempo, alongside qualitative elements such as performance instructions. This multi-dimensionality supports essential MIR tasks like interval detection and rhythmic grouping while also allowing for nuanced analysis and playback. However, symbolic systems must balance precision with flexibility, as traditional notation often only approximates the musical experience [74].

Data abstraction is critical in symbolic representation systems, enabling musical features to be encoded independently of specific units, such as Hertz for pitch

2.3. Music Representation

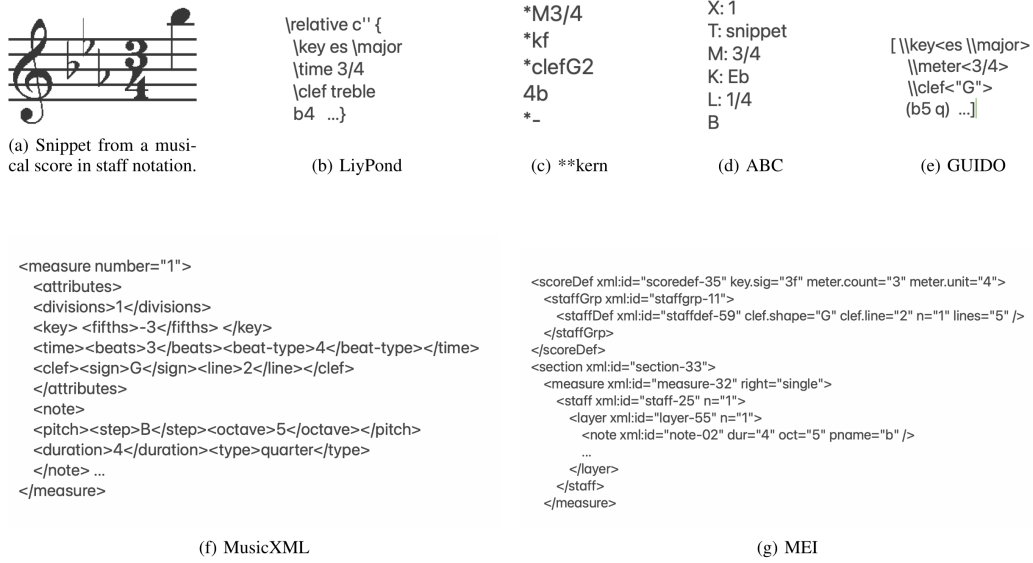


Figure 2.12: Symbolic formats illustrating a B5 note in the key of Eb with a time signature of 3/4. Subplots include: (a) score representation, (b) LilyPond, (c) **kern, (d) ABC, (e) GUIDO, (f) MusicXML, and (g) MEI.

or seconds for time [405]. This abstraction facilitates generalised musical operations foundational to MIR tasks and ensures that symbolic representations can be consistently interpreted across applications.

Representing musical time and tempo in symbolic systems also poses challenges, as these elements need to balance the expressive temporal dynamics essential to music with the consistency required for reliable representation. Typical approaches use symbolic time abstractions that retain the key relationships among musical events, though synchronising real-time with metrical time remains an active research area [83].

For MIR and musicological applications, MRSs allow efficient sharing, retrieval, and collaborative analysis of music data. These systems range from general-purpose formats to specialised systems supporting tasks such as recording, analysis, or generative applications [404]. In this work, we examine MRSs that facilitate computational and analytical approaches, focusing on applications in MIR, musicology, and Deep Learning (DL) systems. Figure 2.12 presents the symbolic representations discussed in this work, showcasing diverse syntaxes used for encoding music data.

Musical Instrument Digital Interface (MIDI) is a protocol developed to facilitate digital communication between musical instruments and computers [342].

Initially introduced in the early 1980s, MIDI was conceived as a means to control electronic musical instruments remotely and in real-time, primarily by encoding performance information rather than actual sound [342, 13].

This standard is based on two basic messages: *note-on* and *note-off*, which encode with them details such as pitch, velocity, and timing, thereby functioning more as a digital score than an audio recording. MIDI files, formalized through the Standard MIDI File (SMF) format in 1988, allow for the exchange of MIDI data across different systems, preserving not only musical notes but also tempo and time signatures, which are vital for synchronization [106].

Despite its widespread use, MIDI faces limitations in its representation capabilities. For instance, MIDI encodes pitch based on a fixed, equal-tempered scale, and does not accommodate microtonal variations, which can be crucial in non-Western musical contexts [252]. Moreover, the protocol’s focus on performance data—such as dynamics and articulations—makes it ideal for applications requiring control over musical expression but less so for those needing detailed notational information, like key signatures, chords, or structural details [22]. However, MIDI’s structure, which organizes data into a series of timed events rather than a continuous stream, allows for intricate manipulations of musical timing, albeit at the expense of more nuanced musical descriptions such as phrasing or the subtleties of harmonic relationships [405, 282, 241].

MusicXML stands as a pivotal format in the digital music domain [159], encapsulating the intricacies of musical notation within a universally transferable XML-based framework. Initially designed to serve as an online sheet music and music software exchange medium, MusicXML was aimed to do for music notation what MIDI did for electronic musical instruments. Unlike MIDI, which is primarily focused on performance data, MusicXML covers a broader spectrum, including both the visual representation of music scores and detailed encoding of musical elements like pitch, duration, and dynamics, thus ensuring both human and machine readability [282].

Rooted in the Extensible Markup Language (XML), MusicXML allows for a detailed representation of almost every musical element and offers flexibility in terms of data hierarchy, supporting both part-wise and time-wise score representations. This flexibility is enhanced by the use of XSLT³, which can alternate between hierarchical representations to simulate the structure of musical scores.

³<https://www.w3.org/TR/xslt-30/>

Such capabilities make MusicXML particularly suited for applications that span musical notation, performance, analysis, and retrieval.

Despite its strengths, MusicXML is not without its limitations. The format is complex and verbose, which can introduce challenges in data processing and interpretation. While it excels in notational accuracy and interoperability between various music software, its application in training DL models for music is limited. Moreover, encoding or decoding MusicXML can be cumbersome for AI models that thrive on larger datasets and longer context lengths [252].

Music Encoding Initiative (MEI) is a community-driven, open-source endeavour to define a system for encoding musical documents into a machine-readable format [174]. Developed to preserve and share detailed information about musical notations, MEI utilizes an XML-based schema to encode both the intellectual and physical properties of music notation documents, thereby facilitating consistent search, retrieval, display, and exchange of musical data across various platforms [252].

The core of MEI is structured into 23 modules, each designed to encapsulate unique characteristics of musical source encoding, expressed in an XML meta-schema language known as the “One Document Does-it-all” (ODD) format. This modular and extensible structure supports encoding a diverse range of music notation systems beyond just common Western notation, including mensural (Renaissance) and neume (Medieval) notations .

In contrast to MusicXML, which primarily facilitates interchange between notation editors, one of MEI’s primary goals is to create a semantically rich model for music notation that goes beyond mere visual imitation to preserve the unique structure and semantics of each notation system. This enables the encoding of traditional facsimile, critical editions, and performance editions, promoting the development of extensive and international archives of notated music which serve as crucial resources for music editions, performances, analyses, and research [174].

However, the complexity, focus on detailed musical notation and emphasis on visual representation rather than audio features, the large data files, and limited integration with Machine Learning (ML) tools make MEI less suitable for DL applications that require more streamlined and uniform data formats [252].

LilyPond is a music notation software that emphasizes the creation of visually pleasing sheet music via a high-level description file [290]. It is similar to LaTeX in its approach, allowing users to input musical notation in plain text format

which is then compiled into professionally engraved scores. Developed with a comprehensive syntax that allows to cover a wide range of musical symbols and formatting options, LilyPond is capable of handling complex scores that require professional-quality output [252].

The software functions by reading and processing files that contain formal representations of the music, outputting to formats such as PostScript or PDF. The pitch in LilyPond is indicated using lowercase letters, and octave adjustments are made with apostrophes (') or commas (,), with each symbol raising or lowering the pitch by one octave respectively. Alterations to pitch such as sharps and flats are added by appending '*is*' for sharps and '*es*' for flats to the note name. Durations are noted by their reciprocal values; for example, a quarter note is represented as '4' and a half note as '2' [290].

While it offers extensive control over musical notation, LilyPond's complexity and focus on visual presentation can introduce challenges when being employed for MIR tasks, particularly in tasks like automatic transcription of music [252].

****kern** is a symbolic music representation system developed as part of the Humdrum Toolkit, created by David Huron in the 1980s [204]. This system is designed to encode Western music notation efficiently, focusing on common practice music notation and is used for computational music analysis. The format represents pitch, rhythm, meter, and articulation in a clear and straightforward syntax that is both human-readable and easily processed by computers.

The ****kern** format is notable for its extensibility and flexibility, enabling users to include detailed musical parameters and metadata, which is crucial for symbolic music analysis, pattern recognition, and music generation. Models using ****kern** can analyse musical structures, recognize stylistic features, and generate compositions that comply with specific musical conventions [2].

In ****kern**, pitch information is represented using a combination of upper and lowercase letters that denote octaves: for instance, the lowercase 'c' represents Middle C (C4), while multiple 'c's (e.g., 'ccc') indicate higher octaves, and uppercase 'C' denotes lower octaves (e.g., 'C' for C3). Accidental symbols are also incorporated, with '#' indicating a sharp, '-' a flat, and 'n' a natural note. Durations are marked numerically, with '4' for a quarter note and '8' for an eighth, among others [204].

While ****kern** is highly effective for notational accuracy and computational analyses, its focus on the notational aspect may omit some expressive details

found in performance data, such as dynamics and exact timing nuances [252].

ABC Notation was developed by Chris Walshaw in 1997 and is the de facto standard for folk and traditional music notation, offering a simple, text-based approach to represent music⁴. It is written using ASCII characters including letters, digits, and punctuation marks, which makes it highly accessible and widely adopted for sharing music online. ABC notation consists of two parts: a header that contains metadata such as the tune’s title, meter, default note length, key, and reference numbers, and a body that describes the actual musical content—notes, rests, bars, and other musical symbols.

The notes in ABC are denoted using the English note names, with uppercase letters (A–G) representing the lower octave and lowercase letters (a–g) for the higher octave. Octave modifiers include the comma (,) to indicate lower octaves and the apostrophe (') for higher octaves. For instance, “C,” represents a low C, while “c'” indicates a high C. Rests are denoted by 'z' or 'x', and their duration can be modified similarly to notes. Musical nuances such as dynamics, articulations, and decorations are expressed using textual expressions enclosed in exclamation marks (e.g., !trill!) .

ABC’s structure allows for straightforward conversion between ABC and other music notation formats, notably MIDI. This facilitates its use in a wide range of applications, from educational tools to software that performs complex tasks like automatic transcription and music generation. For example, the software package ABC Music includes tools like *abc2midi*, which converts ABC files to MIDI, supporting features such as multivoiced files, guitar chord expansion, voice transposition, and percussion accompaniment.

Despite its simplicity, ABC notation’s ability to encode detailed musical information makes it suitable for computational musicology and the development of music Artificial Intelligence (AI) models. It’s particularly valued in projects that require converting textual descriptions into symbolic music notation, owing to its compatibility with natural language notations. This compatibility enhances the potential of ABC notation in training text-to-symbolic music models, such as those aimed at generating music from textual descriptions [252].

GUIDO Music Notation (GMN) is designed to be a robust, yet straightforward format for encoding musical scores. Developed with simplicity and computational efficiency in mind, GMN offers a flexible method for representing a wide

⁴<https://abcnotation.com/>

range of musical styles and complexities [196, 195]. It uses a clear and concise syntax to denote pitches, durations, dynamics, articulations, and other musical elements. The hierarchical structure of GUIDO allows for well-organized musical information, making it accessible for various digital music applications.

The main advantage of GMN is its balance between human readability and machine processability. This balance facilitates tasks such as symbolic music analysis, composition, and pattern recognition in ML models. These models can leverage GUIDO’s structured format to explore musical structures, generate new compositions, and identify stylistic patterns across genres. Moreover, the readability of GMN aids in debugging and data interpretation during model development and evaluation phases.

However, GMN’s simplicity might come at the cost of omitting detailed expressive nuances, which are often necessary for complex performance analysis and intricate compositions, making it potentially less detailed than formats like MusicXML or MEI.

2.3.5 Knowledge Representation of Music

Given the complex and multifaceted nature of music, knowledge representation offers a structured approach to capture and model its conceptual and relational elements. These approaches go beyond simple data encoding, which focuses on technically structuring music data into formats that are readable and analysable by computer systems. While data encoding aids retrieval and basic analysis, knowledge representation seeks to model deeper, conceptual aspects of music, such as genre, instrumentation, and notation, in a structured format [75]. This distinction underscores that data encoding primarily deals with technical readability, while knowledge representation addresses the complexity of musical interpretation, relational structures, and high-level descriptions that cannot be derived directly from raw audio or symbolic data alone.

The field of knowledge representation has its origins in mid-20th-century logic and artificial intelligence, with early developments like semantic networks and symbolic logic [210]. Today, the Semantic Web represents a prominent tool for advancing these ideas, offering a framework for creating machine-interpretable data on a global scale.

Semantic Web technologies, including ontologies and linked data, play a pivotal role in representing and linking music-related knowledge. Ontologies provide for-

mal definitions of the concepts and relationships within a given domain, allowing data to be tied to its meaning in a standardized way [324]. This approach enables diverse data sources—ranging from manually annotated music scores and editorial data to social and automatically generated content—to be logically interconnected, moving beyond simple text-based searches to context-aware, semantically driven queries [135].

In the field of MIR, these representations are crucial. By enabling the integration of heterogeneous datasets and allowing for sophisticated, context-rich queries, Semantic Web technologies bring new depth to music research, offering a distributed knowledge environment that fosters interoperability and enriches musical analysis [52].

Fundamentals of Semantic Web Technologies

The Semantic Web, pioneered by Tim Berners-Lee, aims to transform the Web from a collection of documents into an interconnected web of data, enabling machines to process and interpret information in ways that enhance interoperability, data integration, and automated reasoning [29]. This transformation relies on standardized frameworks for structuring, linking, and querying data, with Resource Description Framework (RDF), Web Ontology Language (OWL), and SPARQL forming the core technologies essential for creating and managing knowledge graphs and ontologies.

Knowledge Graphs The term KG has a long history, with origins dating back to at least 1972 [350]. However, its modern usage gained traction following Google’s 2012 announcement of the Google Knowledge Graph [363], which was subsequently adopted by major tech companies, including Amazon [233], IBM [112], Microsoft [356], Uber [173], and others.

At the core of KGs is the idea of using graph-based structures to represent and integrate diverse data sources, with the goal of accumulating and conveying knowledge about the world [293]. In KGs, nodes represent entities, while edges represent relationships between these entities, offering an intuitive and flexible abstraction for complex, interconnected data. KGs are particularly advantageous for tasks that require integrating, managing, and extracting value from diverse data sources, as they allow flexible schema design, support handling incomplete data, and are well-suited for domains with intricate relationships, such as social networks, biological data, bibliographic records, and transport systems [11].

Compared to relational databases and some NoSQL alternatives, graph-based data models enable advanced data retrieval and analysis. Graph-specific query languages, such as SPARQL and Cypher, allow not only traditional relational operations but also navigational queries that explore complex, multi-step connections among entities [10]. Standard knowledge representation tools—such as ontologies [193]—can be incorporated into KGs to formalize and reason about the semantics of entities and their relationships, further enhancing their interpretability.

A KG can serve as an evolving, centralized repository of knowledge within organizations or communities [293], with applications spanning between *open* and *enterprise* contexts. Open knowledge graphs—such as DBpedia [240] and Wikidata [398]—are accessible for public use, covering broad domains or specialized areas like media [325], government [186], and life sciences [50]. Conversely, enterprise KGs are proprietary, optimized for commercial use cases such as search optimization (e.g., Google Knowledge Graph [363]), recommendation engines (e.g., LinkedIn [185]) and business analytics.

Ontologies: The Foundation of Structured Knowledge To enable precise inference and automated reasoning within a KG, it is essential to define terms explicitly. For example, when considering events within a dataset, questions arise about the exact meaning of “event”: does an event occur only once, or can it recur? Can an event take place in multiple venues, or should each location be treated as a separate event? Such questions highlight the need for *conventions* that formalize what entities and relationships mean within a particular context, ensuring consistency and interpretability.

In computing, an *ontology* is a structured, formal representation of knowledge that defines entities, their attributes, and their relationships within a given domain [194]. Originating from the field of philosophy, ontology is concerned with categorizing and understanding the nature of entities. In practical terms, an ontology in the context of computer science serves as a blueprint that defines how data should be structured and understood within a specific domain. Going back to the previous example, an ontology might define an “event” as something with exactly one venue and start date, while another might allow multiple venues and start times, capturing different conventions and perspectives [194].

Ontologies are useful for guiding how data is modelled within knowledge graphs. For example, if we choose a strict interpretation of “event” (e.g., one venue, one

date), we may need to split certain occurrences into multiple entries to conform to this definition. In contrast, a more flexible ontology might allow events to encompass multiple occurrences and venues, enabling a different modelling approach. By formalizing these conventions, ontologies allow automated systems to infer additional knowledge based on the relationships and constraints defined within the ontology.

The effectiveness of an ontology largely depends on its level of adoption and how well it aligns with the needs of users and data applications. Consistent adoption within a single knowledge graph can streamline data use and integration, while broad agreement across multiple systems enhances interoperability, allowing knowledge to be shared and reused effectively.

2.4 Multimodality in Music Information Retrieval

Multimodality and Multimodal Deep Learning (MMDL) are gaining significant attention across diverse AI fields, including natural language processing, computer vision, and music information retrieval. In particular, MMDL has been applied to several domains including including biometrics [365], self-driving cars [308], robotics [14] and healthcare [413].

However, while the foundation of multimodality is conceptually similar across these domains, it differs in a fundamental way: in most areas, multimodality is closely associated with human communication—such as spoken language, visual expressions, or written text—aiming to capture and interpret the nuances of human interaction.

In MIR, by contrast, multimodality is not about reproducing human communication but rather encompasses a diverse set of representations of music, each offering unique insights into its structure, performance, and content.

MIR has been inherently multimodal since its inception, as stated by Downie in his foundational paper:

MIR is concerned with the extraction, analysis, and usage of information about any kind of music entity (e.g., a song or a music artist) on any representation level (for example, audio signal, symbolic MIDI representation of a piece of music, or name of a music artist). [119]

However, despite its widespread adoption, the concept of multimodality remains ambiguous, particularly in the context of music data. In this section, we

begin by examining the meaning of multimodality, exploring various definitions and offering a brief overview of its applications in the MIR field.

2.4.1 Towards a Definition of Multimodality in MIR

In the evolving field of Multimodal Learning, a clear understanding of multimodality is essential. However, before examining the specifics of multimodality, it is essential to distinguish it from the concept of *multimedia*, as these terms are frequently misinterpreted.

According to the Oxford Advanced Learner’s Dictionary, “medium” refers to a conduit for communicating information (e.g., text, images, and sounds), whereas “modality” signifies the particular way something is experienced or executed. While multimedia might involve multiple types of media used simultaneously, multimodality focuses on the nature of the experiences these media induce.

Existing definitions of multimodality typically hinge on human sensory experiences, presupposing that humans interact with the world through a multimodal lens—seeing, hearing, feeling, smelling, and tasting [15]. However, this human-centred view presents several limitations:

- *Input Types:* The sensory capabilities of machines can exceed those of humans, who are confined to their natural senses.
- *Input Range:* Human sensors are limited to a specific range of signals, whereas machines can potentially detect an expansive spectrum, bounded only by current technological advancements.
- *Interpretation:* Humans naturally interpret sensory data, whereas machines require specific programming to derive meaning from the signals they process.

Alternatively, a machine-centred perspective might define multimodality in terms of multiple data representations. However, this definition is problematic as it could lead us to mistakenly categorize different file formats like PNG and JPEG as distinct modalities when they are simply different methods of encoding information. Thus, a more refined, task-relative definition is proposed:

Definition. *A machine learning task is deemed multimodal when its inputs or outputs are represented differently or consist of distinct types of atomic units of information. [15]*

2.4. Multimodality in Music Information Retrieval

This definition underscores that multimodality is pertinent when it provides information unattainable through extensive amounts of unimodal data. For instance, if a task transforms spoken language into text and does not require the nuances specific to spoken forms—like intonation or pauses—then it is not considered multimodal under this task-dependent framework.

Adopting the foundational insights from [298], Christodoulou et al. propose a new definition specifically tailored for multimodal music datasets:

Definition. *A multimodal music dataset can be defined as diverse data types that offer complementary insights for a specific music processing task, regardless of source, format, or perceptual characteristics.*

This definition emphasises the utility of data diversity in enhancing music analysis, focusing on the synergy of various data forms to provide a holistic understanding of music phenomena. By decoupling the concept of modality from human sensory modalities and specific data formats, this definition broadens the scope of what can be considered multimodal in music information tasks, facilitating more innovative approaches to dataset construction and usage in computational music analysis.

In a related examination, Simonetta et al. [359] present a detailed analysis of multimodality in MIR. In this context, a modality is defined as a specific method for digitising music information. Different modalities arise from various transducers, various locations or times, and can be tied to different media. For example, a single piece of music may have several associated modalities, such as audio recordings, lyrics, symbolic scores, and album artwork.

Under this definition, multimodal music information processing refers to an MIR approach that takes multiple modalities of the same piece of music as input. This definition is valuable for technical applications in MIR, where it underscores the importance of processing data in various forms and through diverse digitisation methods.

However, while this last definition offers a strong foundation for the technical processing of music data, it can be limiting when applied to broader MIR tasks. As noted in [66], focusing exclusively on the technical aspects of digitisation and multiple modalities overlooks the need for a flexible and integrated approach that considers the specific musical task at hand. Therefore, for multimodal analysis in MIR to be fully effective, it is essential to balance the technical aspects of modality with the functional requirements of the task, allowing for greater adaptability in

handling different types of musical data.

This need for balance is echoed in a previous survey on multimodal MIR, in which Essid et al. [130] introduce the term *cross-modal processing*, a concept that has since been adopted in other works, such as [283]. According to [130], *cross-modal processing* is defined as “the effort of characterising the ‘relationships’ between the different modalities reflecting the content being analysed”. This stands in contrast to *multimodal fusion*, which is described as “the problem of efficiently combining the information conveyed by the different modalities to perform a more thorough analysis of the content.”

These definitions suggest that certain synchronisation algorithms, such as audio-to-score alignment, should be categorised as cross-modal processing.

Of all the definitions presented, we align most closely with those of [66] and [298]. Consequently, we will use the term “multimodal” under the perspectives outlined in these definitions throughout this work, emphasizing the importance of balancing technical aspects with functional requirements and recognizing the complementary insights offered by diverse data types for specific music processing tasks.

2.4.2 Challenges of Multimodal Deep Learning for MIR

MMDL research focuses on developing neural architectures that can effectively integrate diverse modalities by managing both view-specific and cross-view dynamics [327]. View-specific dynamics occur within a single modality independently of others, while cross-view dynamics involve interactions among modalities. For instance, the activation of facial muscles during a smile exemplifies view-specific dynamics, whereas the co-occurrence of a smile with a positive utterance represents a cross-view dynamic [421]. An effective MMDL model in MIR must address several challenges inherent to handling these dynamics [15]:

- **Representation:** Learning machine-interpretable representations for each modality is essential for compatibility with machine-learning models. In MIR, this requires encoding audio, symbolic music data, lyrics, and metadata features in ways that maximize interpretability and relevance for the task at hand [252].
- **Translation:** This involves transforming data from one modality to another, such as converting audio to symbolic representations in music transcription

tasks [22, 360].

- **Alignment:** Recognizing relationships across modalities is critical. In MIR, this could involve aligning lyrics to specific audio segments [145] or synchronizing symbolic notation with corresponding audio excerpts [279], thereby enhancing interpretability through cross-modal connections.
- **Fusion:** Aggregating information from multiple modalities is essential for tasks involving classification or decision-making in MIR. By integrating insights from lyrics, audio, and symbolic representations, systems can achieve greater robustness and accuracy [359].
- **Co-Learning:** This refers to the transfer of knowledge across modalities, enabling models to generalize more effectively. In MIR, co-learning may facilitate shared representations across audio and symbolic data, potentially reducing the need for extensive labeled data and enhancing performance in low-resource environments [154].

In this thesis, we contribute to the field of multimodal learning in two key areas: *alignment* (6.3) and *translation* (6.5), addressing essential aspects of cross-modal integration for exploring possible solution to tackle limits of symbolic music data integration.

CHAPTER 3

Representing Musical Knowledge

3.1 Introduction

Musical heritage encompasses a diversity of human expressions and experiences, leaving heterogeneous traces that are difficult to describe, connect, and preserve [178]. Western music cultural heritage developed through varied sources: musical contents and objects (such as tunes, scores, melodies, notations, recordings, etc.) linked to tangible objects (theatres, conservatoires, instruments, etc.) but also to their cultural and historical contexts, opinions and stories told by people with diverse social and artistic roles (scholars, writers, students, intellectuals, musicians, politicians, journalists, etc.), and facts expressed in different styles and perspectives (memoire, reportage, news, biographies, reviews) in different languages (English, Italian, French, Spanish, and German) and across centuries [41]. This diversity creates unique opportunities as well as challenges for researchers and practitioners attempting to study and preserve music heritage.

However, the fragmented nature of the data limits our ability to fully understand the cultural significance and historical trajectory of musical works. As such,

there is a pressing need for frameworks that can integrate both musical content and context, enabling better preservation, study, and dissemination of music in its full richness.

3.1.1 Challenges and Requirements for Interoperability

Music data can describe two main musical dimensions: *content*, which encompasses intrinsic elements like pitch, harmony, and rhythm, and *context*, which includes broader information such as cultural, historical, and publication metadata.

Music metadata – also known as contextual, bibliographic, or documentary data – plays a crucial role in identifying and describing musical works, their creators, recordings, and performances. For the music industry, metadata is vital for efficiently managing and distributing music, supporting tasks such as search, recommendation, and cataloguing [296]. Accurate metadata ensures that artists are properly credited and compensated [364], and for musical heritage, metadata facilitates the preservation and dissemination of works across different cultural and historical contexts [156].

In this regard, metadata also promotes diversity and inclusivity by highlighting lesser-known genres and artists, thereby fostering a more comprehensive understanding of global musical traditions [105]. However, challenges arise due to the inconsistency and fragmentation of metadata across various systems and musical traditions, each with unique conventions for describing elements like composers, performers, or works. For example, the term “composition” in classical music may diverge from the concept of “track” or “song” in popular music, leading to fragmentation that hampers interoperability between systems and datasets.

Alongside metadata, musical content itself – whether in the form of symbolic data (such as music scores, or MIDI files) or audio recordings – poses its own challenges for computational representation (see Section 2.3). Audio data is unstructured and requires extensive feature extraction, while symbolic data is more structured but often not as expressive as audio signal [396]. Moreover, numerous audio and symbolic representations have been proposed, each tailored to specific tasks or genres. However, interoperability among these representations is often limited. This has led to significant data fragmentation issues in the MIR field, with small datasets structured in varied formats to suit specific tasks and applications. This fragmentation poses two main challenges: first, it demands substantial

resources for collecting, harmonising, and preprocessing disparate data; second, it hinders comparison and reproducibility, as findings from different datasets are often misaligned. As a result, achieving data interoperability is increasingly critical. Effective data integration must go beyond syntactic alignment (i.e., standardised formats and structures) to also ensure semantic consistency, preserving the meaning and relationships of musical elements across representations.

Domain specificity hampers interoperability

Existing ontologies for music data are typically tailored to specific use cases and requirements, limiting their general applicability.

For metadata, Music Ontology (MO) [324] leans towards modelling discographic data with a focus on contemporary music, whereas DOREMUS [64] is inherently rooted in classical music. Nevertheless, when drifting from discographic data and classical music, or attempting to reuse both models, addressing e.g. cultural heritage requirements while fostering interoperability becomes difficult. Indeed, a model reflecting the view and the interpretations ascribable to a musical genre, stakeholder, or dataset type may be difficult to reuse and extend to other domains. For instance, a music artefact may originate from oral transmission or be the result of a creative process that does not necessarily entail a formal composition process. The latter is common in songwriting, but also in folk music whenever a set of tunes (collected from different manuscripts) allows for the identification of a tune family [392]. Similarly, when expressing relationships between musical artefacts (alias derivations), it is important not to impose any modelling bias that may constrain possible interpretations (e.g. an arrangement having proper musical identity vs simply providing a different instrumentation). This is commonly referred to as “dominance of concept” [64], whose definition should be left to users depending on their data and domain expertise.

The same limitations apply to ontologies designed for music content data, which are often created to represent specific types of musical information. For example, some ontologies focus on the semantics of music notation [213], while others target data from particular representation systems, such as MIDI [272]. These focused approaches hinder interoperability between systems and formats, as they lack a flexible structure for integrating diverse types of musical content.

Rather than attempting to achieve consensus on musical concepts and jargon, accounting for the interoperability calls for an abstraction layer for music data

(“*zoom-out*”) that can then be specialised, extended, and adapted to address domain-specific requirements (“*zoom-in*”).

Expressivity is needed at different levels

Another requirement for interoperability and reuse across various data sources is providing expressivity at different degrees, i.e. the possibility to conveniently describe music data at the right level of detail. For example, one data source may have granular/detailed information that requires high semantic expressivity (a composition process spread over different time, places, and involving more artists); whereas others may have basic (only the name of an artist is known) or even incomplete and uncertain information (a composition tentatively attributed to an artist).

For metadata, the WikiProject Music¹ has been successful in providing expressivity to represent music metadata from different sources. As an extreme case of ontological flexibility, the schema underlying Wikidata – an open-ended, multi-domain KG built collaboratively like Wikipedia – is not specified in a previously agreed ontology, and the high expressivity overly adds complexity to the model. This is due to Wikidata’s scope being the most general.

Provenance is fundamental for data integration

Accounting for provenance is a central requirement for both cultural heritage and music industry. This becomes fundamental when integrating KGs from different datasets and stakeholders – as every single bit of data (each triple) should be attributable to a dataset/KGs. Furthermore, integrating provenance is also needed within the context of a single dataset, at least for claims and links.

Claims-Interpretations Cultural heritage applications often require representing debatable statements or claims [86, 87]. These are usually the result of an interpretation process based on factual or documentary evidence (a dataset, a manuscript, etc.), and following a methodology and/or theory. Examples include personal information (e.g. the year/place of birth of a composer), and authorship claims (e.g. a composition being attributed to an artist).

Links and identifiers These includes links to artists’ official websites, fan pages, discussion forums, music reviews, record shops; as well as identifiers from mu-

¹https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Music

sic databases (e.g. MusicBrainz, Discogs, AllMusic), streaming platforms (e.g. Deezer, Spotify), and authoritative sources (e.g. ISNI, ISWC, ISRC). As most links and identifiers are crowdsourced or automatically inferred by entity linking algorithms, modelling provenance here promotes traceability and accountability of data sources.

3.1.2 Towards a Unified Model for Music Representation

The ontology engineering efforts described in this chapter have been conducted within the framework of the H2020 Polifonia project². The Polifonia project brings together memory institutions, museums, music archives, scholars, commercial organisations, and citizens who ask complex questions (e.g. “*Which tunes share melodic patterns and geographical origin?*”; “*How do libretto and music relate, e.g. in describing an emotion?*”; “*Can we trace the evolution of tonality and transition from modal to tonal?*”) across these multi-perspective and multi-modal sources. This demands the integration of musicological (notes, chords, modes, theories), historical (events, persons, places, objects) and archival/preservation (metadata, descriptors) data and perspectives. The project comprises 4 cultural institutions (CNAM, NISV, MiC, KNAW) and 10 pilots with a large variety and number of requirements. Ontologies and KGs have the potential to overcome these challenges, and shed light on this wealth of resources by extracting, materialising and linking new music history knowledge that was previously overlooked and therefore missing [277, 43].

Although various ontologies have been proposed to model some aspects of musical heritage interest, they are individually insufficient to overcome the challenge of integrating the notation, metadata, and historical contexts needed for multi-perspective cultural analyses; thus leaving questions about the relationship between musical theory (melodies, tonalities, chords) and culture (historical events, architecture, geography) unanswered. To date, no available ontological framework integrates music metadata, notation, annotation, source provenance, and cultural heritage object descriptions. To the best of our knowledge, no toolkits exist to support knowledge engineering tasks around the lifecycle of competency questions, which is a central project requirement given the large number of variety of stakeholders, pilots and questions.

²<https://polifonia-project.eu/>

3.1.3 Our contribution

In this chapter, we describe the Polifonia Ontology Network (PON), a set of new ontologies formalising the semantics of music representation, metadata, annotation, analysis, mediums of performance (instruments), and historical sources (provenance), enabling the creation of interoperable knowledge graphs from music datasets. These ontologies address **RQ1** (c.f. Section 1.1.1) by establishing a unified framework for representing music-related data, enhancing interoperability and paving the way to the creation of harmonised KGs of musical data. The contributions detailed in this chapter have been published across several peer-reviewed works [94, 88, 52, 27].

To achieve this, we apply and extend XD [36], a well-known ontology design methodology where ontological requirements are gathered from a comprehensive inventory of CQs, and modularity is fostered through the reuse of ODPs [151, 192]. We also release the PolifoniaCQ dataset, a collection of 361 competency questions on musical heritage. Further, we validate PON and provide evidence of its current and planned (re)use by three different types of users: (i) the Polifonia pilots, using them to generate musical culture KGs; (ii) a number of industrial and institutional stakeholders and early adopters, planning to use PON to annotate their in-house datasets; and (iii) a survey run in the Semantic Web and Music Technology communities showing intentions of use.

More specifically, the contributions of this chapter are as follows:

- *Extensions to XD* centred around CQ extraction and enhancement, including both methodological (a CQ-elicitation framework to mirror use cases from domain experts) and technological (a toolkit for assisted design and iterative improvement of CQs through language models) aspects (Section 3.3).
- *PolifoniaCQ*, a new dataset of competency questions driving the design and the evaluation of PON, with associated stories and personas (Section 3.3.1).
- The *Polifonia Ontology Network (PON v1.0)* resources, available on GitHub³ and including 15 (CC-BY 4.0) ontology modules (Section 3.4).
- *Evidence of reuse* and impact from music stakeholders, applications within Polifonia, and interest from various research communities (Section 3.6).

³<https://github.com/polifonia-project/ontology-network>

- *Example-driven validation* of the model, focused on the data elicited from four different stakeholders.
- *Code support* to create Music Meta KGs without expert knowledge of the model, with automatic alignments to the MO, DOREMUS, and Wikidata.

3.1.4 Chapter Structure

The chapter begins with a review of related works in the field of knowledge engineering for music, also including a discussion on methodologies and workflows for ontology engineering (Section 3.2).

Section 3.3 outlines the methodology adopted for this work, which encompasses two key processes. First, requirement collection (Section 3.3.1) details the process of gathering the needs and expectations of various stakeholders involved in the Polifonia project. Second, Ontology Network Design and Development (Section 3.3.2) describes the design choices and development processes for creating a robust and scalable ontology network (PON) that integrates both music content and metadata.

Section 3.4 provides an in-depth description of the Polifonia Ontology Network, focusing on its structure, the modules it incorporates, and how it supports the integration of diverse types of music data.

A dedicated focus is given to the Music Meta Module (Section 3.5), which is specifically designed to handle the diverse and complex metadata requirements of musical heritage.

Section 3.6 reviews the adoption and reuse of PON within and beyond the Polifonia project. Evidence of its use in Polifonia pilots is presented in Section 3.6.1, while Section 3.6.2 summarises interest in PON reuse based on a survey of the Semantic Web and Music Technology communities. Section 3.6.3 describes early PON adoption by Polifonia Stakeholder Network members and resulting synergies for validating and annotating cultural and industrial datasets.

The chapter concludes with Section 3.7, which summarises ontology engineering achievements, discusses ongoing challenges, and outlines future directions for PON.

3.2 Related Work

Ontologies play a fundamental role in the representation and management of knowledge, by providing common vocabularies to describe resources and queries.

Several ontologies exist in the music domain for addressing diverse applications, dealing with both music content and metadata at different levels of specificity. MusoW [85] is a catalogue indexing online music resources, including ontologies and KGs. Here, we focus on music ontologies and categorise them according to their reference domain: (i) metadata; (ii) music theory; (iii) music notation; and (iv) audio features.

In this section, we explore the key areas of ontology engineering in music, beginning with ontologies for describing both the context and content of music. Table 3.1 presents a taxonomy of music ontologies, categorised by their domain, scope, and the year of the latest release, providing an overview of the diverse landscape of existing models. Next, we examine the primary methodologies that have been developed for ontology engineering, focusing on their evolution from early frameworks to more collaborative and agile approaches.

3.2.1 Ontologies for Describing Music Context

Ontologies such as the *Music Ontology (MO)* [324] and *DOREMUS* [249] play a foundational role in describing high-level metadata about musical works, composers, and performances. When looking at these ontologies, MO leans towards modelling discographic data with a focus on contemporary music, whereas DOREMUS is inherently rooted in classical music. These ontologies have been demonstrated to model metadata from MusicBrainz and BBC Music [326], and from classical music libraries and radio broadcasts for concerts programming [64], respectively. Their specificity makes them appealing when downstream applications show considerable overlap in terms of requirements and data. Examples include the reuse of MO in the WASABI project [45], to support the semantic annotation of audio music (emotions, lyrics, structures), but also for music recommendation [337] and listening [4]; and the adoption of DOREMUS by *Philharmonie de Paris*, *Bibliothèque Nationale de France*, and *Radio France*. Nevertheless, when drifting from discographic data and classical music, or attempting to reuse both models, addressing e.g. cultural heritage requirements while fostering interoperability becomes difficult.

More specialised ontologies like the *OMAC Ontology* [347] provide an in-depth description of musical claims and interpretations, which are essential for musicological research. Other ontologies focus on modelling emotional responses to music. For instance, the *COMUS Ontology* [334] captures emotional states by integrating both contextual factors and user preferences, while *UniEmotion* [216] classifies tags into positive emotions, negative emotions, and factual descriptors, offering a structured approach to emotion-based music categorisation.

Additionally, the *Performed Music Ontology* focuses on capturing detailed information about live performances, while the *OnVIE Ontology* [377] extends this to the mediums used in musical performances. The *Musical Instrument Taxonomies* [224] and the *Smart Music Instrument Ontology* [389] further contribute by providing conceptual models for the classification and description of instruments, especially in the context of the Internet of Musical Things [388].

Ontologies like the *ArCo ontology* [54] are also crucial for connecting music with its cultural and historical significance. By situating music within broader contexts of cultural heritage, such frameworks allow researchers to examine how different periods and styles influence musical creation and perception. Other related efforts include [43, 206], which aim to integrate music data into larger heritage databases.

Despite their contributions, existing context-related ontologies face challenges in terms of scalability and interoperability. Many models, while rich in metadata, struggle to integrate with other ontologies due to a lack of standardised alignment practices. The Polifonia Ontology Network addresses these issues by aligning context models with existing web resources while ensuring that provenance information is consistently maintained across datasets [52].

3.2.2 Ontologies for Describing Music Content

In addition to context, other ontologies were modelled to capture intrinsic musical properties like music theory, audio signal features, and notation, serving as essential tools for computational analysis, musicology, and creative applications in MIR.

Some ontologies describe different elements ascribable to *music theory*. The *Music Theory Ontology (MTO)* [332] provides a detailed framework for encoding theoretical music concepts, allowing researchers to model elements such as harmony, tonality, and rhythm. More specific ontologies, like the *Functional Harmony Ontology* [214], take this further by reasoning about harmonic sequences

Table 3.1: *Taxonomy of music ontologies based on their domain, scope, and the year of the latest release.*

Ontology	Prefix	Description	Domain	Scope	Last up.	IRI	Reference
Music Ontology	mo	Describing musical artefacts for cataloguing purposes (e.g. artists, releases and tracks).	context	high-level	2013	http://purl.org/ontology/mo/	[324]
Chord Ontology	chord	Provides a common, versatile vocabulary for describing chords and chord sequences following the Harte notation.	symbolic	mid-level	2007	http://purl.org/ontology/chord/	[135]
Tonality Ontology	tonality	Provides high-level and low-level descriptors for tonal content in RDF.	symbolic	mid-level	2008	http://purl.org/ontology/tonality/	[135]
Music Instrument Taxonomies	n/a.	Two interpretations, hence two taxonomies, to categorise musical instruments.	context	mid-level	2011	http://purl.org/ontology/tonality/	[224]
Segment Ontology	n/a.	An ontological framework to annotate music to perform musicological analysis of segments in a piece.	symbolic	mid-level	2011	http://purl.org/ontology/tonality/	[138]
MusicOWL - Music Score Ontology	miso	A comprehensive vocabulary for annotating music scores for melodies, dynamics, and tonalities.	symbolic	low-level	2017	http://linkeddata.uni-muenster.de/ontology/musiscore	[213]
Music Theory Ontology	mtlo	An extension of music ontologies to include the "missing" theoretical concepts-information.	symbolic	low-level	2018	http://purl.org/ontology/mtlo/	[332]
Temperament Ontology	tm	Models the main concepts, relationships, and parameters of musical temperament, and facilitates the description and inference of various characteristics of specific temperaments.	symbolic	mid-level	2011	http://purl.org/ontology/temperament/	[385]
Music Notation Ontology	nm	A general ontology for representing notated music, with MEI support.	symbolic	low-level	2017	http://cedric.cnam.fr/1s1d/ontologies/MusicNote.owl	[62]
COMUS	comus	Emotion state from context and user preference information	context.	high-level	2009	http://ceai.ajou.ac.kr/ontology/0.9/comus.owl	[334]
DOREMUS Audio Features Ontology	mus afo	An extension of the FRBRoo model for describing music catalogs. Common concepts to represent some features of audio signals. Notions of music structures are also present, as well as music theoretic concepts.	context audio	high-level mid-level	2017 2016	http://data.doremus.org/ontology	[249], [2] [7]
Studio ontology	(several)	An extension of the Music Ontology for music production in a recording studio, collecting the following more specific ontologies: audio recording, audio mixing and editing.	audio	mid-level	2011	https://w3id.org/afo/onto/1.1	[134]
ETree	etree	Describing performances and related audio tracks	audio	high-level	2014-2017	https://etree.linkedmusic.org/vocab/	[19]
CALMA Ontology	calma	Describing the relation between an audio track and a blob containing features analysis.	audio	low-level	2014-2017	http://calma.linkedmusic.org/	[18]
The Audio Effects Ontology	afx	Describes audio effects in music production workflows. Can be seen as part of the Studio Ontology.	audio	high-level	2013	https://w3id.org/afx/ontology/1.0#	[408]
CHARM Ontology	charm	Representing hierarchical structures from symbolic music.	context	low-level	2015	https://w3id.org/charm/ontology/1.0#	[176]
Unifmotion: the Emotion Ontology	n/a.	Categorises tags into positive emotional tags, negative emotional tags, and factual tags.	context	high-level	2013	https://w3id.org/unifmotion/ontology/1.0#	[216]
MIDI Linked Data Cloud	midl	Models MIDI streams at the event level.	symbolic	low-level	2017	http://purl.org/midl-ld	[272]
Mobile Audio Ontology	n/a.	A semantic audio framework for the design of novel music consumption experiences on mobile devices.	audio	mid-level	2014	https://w3id.org/mobileaudio/ontology/1.0#	[382]
HaMSE Ontology	hamse	Models symbolic and audio data for multimodal music data processing.	symbolic audio	mid-level	2022	http://purl.org/ontology/hamse/	[314]
Music Note Ontology	mnnot	A general ontology for representing music notes and note-based structures.	symbolic	low-level	2021	http://purl.org/ontology/musicnote/	[315]

and their relationships, while the *Chord Ontology* and *Tonality Ontology* [135] focus on particular aspects such as chords and tonalities, respectively. In addition, the *Segment Ontology* [138] supports musicological analysis by providing an ontological framework to annotate and analyse musical segments, while the *Temperament Ontology* [385] facilitates the description of musical temperament and its associated parameters. The *Diatonic-Chromatic System Ontology* [169] uses reasoning to infer if a score can be classified within the analytical framework of Michael Praetorius (1571–1621).

Ontologies have also attempted to describe *musical notation* and *symbolic representations*. For instance, the MIDI Linked Data Cloud [272] proposes a way to connect symbolic music descriptions that are encoded in the MIDI format. Meanwhile, the CHARM ontology [176] is focused on representing musical structures. The Music Theory Ontology (MTO) [332] aims to capture the theoretical concepts related to music compositions, while the Music Score Ontology (Music OWL) [213] and the Music Annotation Ontology [62] represent the content of a music score. Additionally, the *CHARM Ontology* [176] provides a hierarchical structure for representing symbolic music, aiding in the analysis of complex musical forms.

Other works focus on audio signals or the procedures used to produce them. For example, The Audio Features Ontology [7], The Studio Ontology [134], and The Audio Effects Ontology [408] are dedicated to describing different aspects of audio production. The Computational Analysis of the Live Music Archive (CALMA) [18] project aims to link metadata of music tracks with computational analyses of recordings, through feature extraction, clustering, and classification. Additionally, ontologies have been used to model listeners' habits and music tastes, as well as similarities between different musical pieces [334, 216, 208, 382]. The *Mobile Audio Ontology* [382] extends this work by offering a semantic audio framework designed for novel music consumption experiences on mobile devices.

Other ontologies, such as the *HaMSE Ontology* [314] and the *Music Note Ontology* [315], focus on linking symbolic music representations with audio data, reflecting recent advancements in multimodal music data processing. These initiatives tackle the persistent challenge of unifying symbolic and audio content, offering a more holistic approach to music analysis [91].

Despite the numerous contributions, the scope of these ontologies is often too specific or ingrained in a genre, style, historical period—often addressing individ-

ual music stakeholders and/or datasets. Several ontologies were also developed independently, with little coordination across relevant contributions. In turn, this often hampers reuse and extension, while jeopardising interoperability—an essential requirement for the integration of music datasets [52].

3.2.3 Ontology Engineering Methodologies

Various methodologies have been proposed for ontology engineering over the years, with a recent shift towards collaborative ontology development [361]. This section provides an overview of the key methodologies in ontology engineering.

Early works, such as those by Uschold and King [391], introduced a foundational framework for ontology construction, including steps for defining the purpose, capturing and coding the ontology, integrating existing ontologies, evaluation, and documentation. However, they did not emphasise the use of Competency Question – a tool for collecting requirements that the ontology must address.

METHONTOLOGY [137] expanded on these early efforts by introducing a more structured process involving specification, conceptualisation, formalisation, implementation, and maintenance. While built on existing methods, this approach did not provide any best practices for reuse and integration of existing ontologies.

The *DILIGENT* methodology [311] introduced a collaborative approach by involving domain experts, users, and ontology engineers throughout the ontology lifecycle. This approach introduced iterative feedback and updates but lacked specific design guidelines and was not test-driven. The process starts with building an initial ontology, which users can locally adapt. Feedback is collected and reviewed by a control board before releasing a new version. The local ontologies are then updated accordingly.

More recent methodologies, such as NEON [372], offer more flexibility by supporting iterative and agile development. NEON proposes nine scenarios for ontology development, covering various combinations of reusing, re-engineering, merging, and localising existing resources. It introduces two ontology network life cycle models to manage the development of interconnected ontologies.

SAMOD (Simplified Agile Methodology for Ontology Development) [306] is another agile methodology, which promotes iterative development through small, manageable steps. The process consists of three phases: 1. developing a “modelet” that formalises a subdomain based on a motivation scenario, 2. merging the new modelet with the existing ontology, and 3. refactoring the model. New

milestones are released once all tests are successfully passed. SAMOD emphasises self-explanatory entity names and reusing existing ontologies but lacks explicit guidelines for requirements elicitation and testing.

The most relevant approach to our work is the eXtreme Design (XD) [36, 35], which focuses on reusing Ontology Design Patterns (ODPs) – small, reusable solutions to recurring modelling problems. This agile, iterative methodology involves multiple teams and is heavily test-driven.

3.3 The eXtreme Design Methodology in Polifonia

The Polifonia Ontology Network (PON) addresses the aforementioned challenges to music data representation and integration by integrating heterogeneous requirements related to musical content and contexts into a modular yet unified architecture. To develop PON, we rely on, and extend, the eXtreme Design (XD) [36, 35] ontology engineering methodology. XD fosters the reuse of Ontology Design Patterns (ODPs) [151, 192] and provides support to incrementally address small sets of requirements formalised as Competency Questions (CQs). This minimises the impact of changes in future releases, which is beneficial to Polifonia (heterogeneous project requirements and participants). Moreover, XD has been successfully applied to the cultural heritage domain [55], and our ontology designers have relevant experience in using this methodology.

The application of XD iterates over a series of steps, for which we detail their process while highlighting our main extensions (see Figure 3.1).

3.3.1 Requirements collection

Ontological requirements are collected from *customers* in the form of *user stories* (e.g. “*Tosca* was performed in Rome on 14 January 2000”), which are then translated as CQ – the natural language counterpart of structured queries that the resulting KG should answer [167]. For instance, the previous story example may become “*Where was a musical piece performed?*”.

We borrow techniques from User eXperience design [166] to extend this framework with 3 new sections in the story template: persona, goal, and scenario. The *persona* is a research-based description of a typical user: name, age, occupation, skills and interests. The *goal* is a short textual description of what the persona aims to achieve in the story, complemented by a list of keywords (maximum 5)

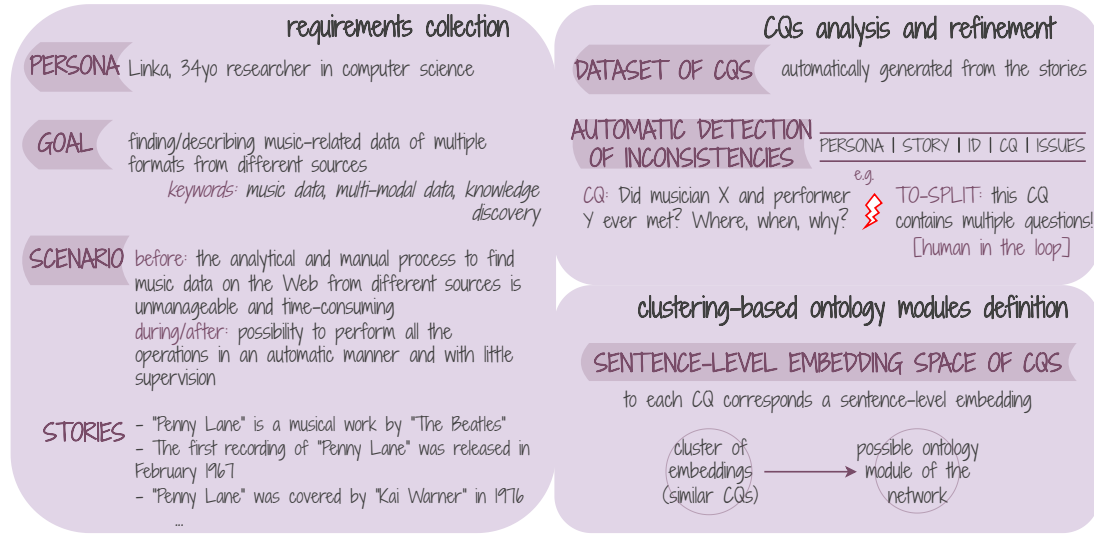


Figure 3.1: Summary of the main Polifonia extensions to the eXtreme Design methodology [318].

provided by the customers. The *scenario* describes how the persona’s goals are currently solved, to contextualise the gap with the resource being developed.

In cooperation with the domain experts in Polifonia (music historians, librarians, computational musicologists, music analysts, archivists, data engineers, etc.), 22 personas have been created⁴ from this step.

Iterative refinement of CQs. Competency questions were then analysed to identify any inconsistencies that could create obstacles for ontology design. Common inconsistencies were due to vague concepts, for instance, the assertion of two compositions being *connected* without any specific context (in terms of the property) on which the connection can be established (e.g. similar melodies, rhythm). Other CQs were found to be overly complex or nested – entailing more than a single requirement as a result of nested logical operators articulating the question (e.g. “*How is track B connected to C to conclude D?*”). Such CQs needed to be conceptually simplified before being processed further.

To efficiently address these inconsistencies, we developed the *Infer, DDesign, CreAte* (IDEA) framework: analytical tools for CQ-driven ontology design based on language models⁵. IDEA automatically extrapolates and organises CQs from a source repository, analyses them to find inconsistencies and similarities, and visually projects them to a sentence-level embedding space [366]. The framework

⁴<https://github.com/polifonia-project/stories>

⁵<https://github.com/polifonia-project/idea>

3.3. The eXtreme Design Methodology in Polifonia

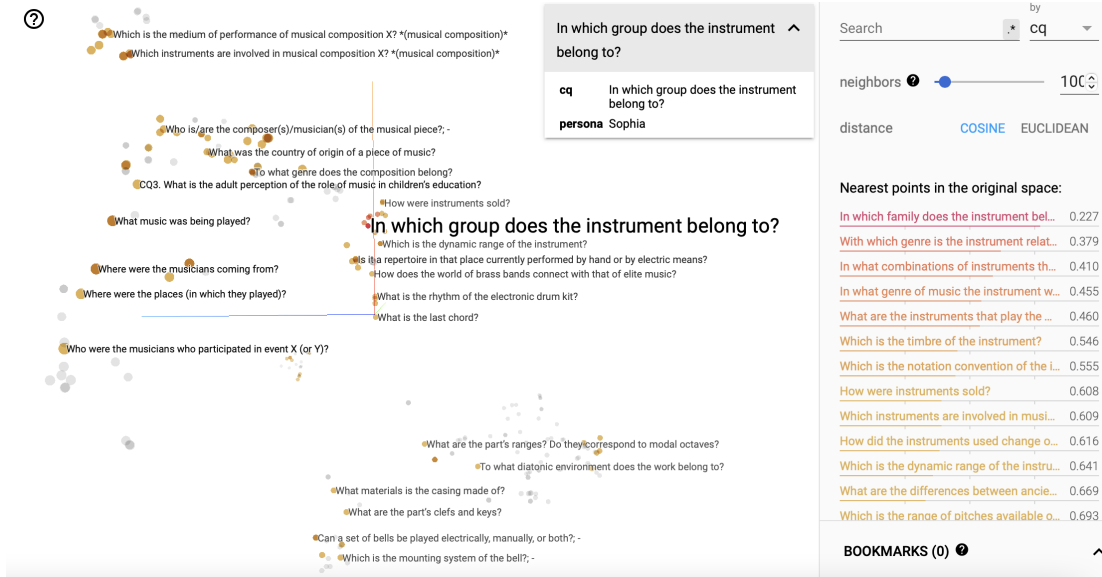


Figure 3.2: Visualisation of the Polifonia CQ embeddings using TensorBoard.

has enabled the iterative refinement and improvement of CQs through human-machine collaboration: questions are first extracted and preliminary validated by tagging them (complex, nested, ill-formed, passing), then brought to the attention of the corresponding ontology designer whenever their intervention is needed. To date, 3 cycles of CQs improvement have been completed with IDEA. Instead, the analysis of CQs embeddings through similarity facilitated the identification of overlapping requirements from the pilots (beyond the syntactic level); which in turn enabled and fuelled discussion from various experts and pilots during our ontology design meetings (e.g. 2 CQs may have similar interpretation or semantics for ontology design, but entail different semantics across pilots).

The PolifoniaCQ dataset. At the end of this process, we obtained 361 CQs, which are systematically collected in the PolifoniaCQ dataset with pointers to their personas and stories. We make this dataset available under CC-BY 4.0⁶.

3.3.2 Ontology Network Design and Development

Clustering CQs as ontology modules. The refined CQs could then be translated in clear, atomic and consistent ontological requirements. Given the wide diversity of CQs – ranging from general events to musicological interpretations of specific passages in compositions, the first step was to achieve a meaningful categorisation into thematic clusters. This step led to the definition of the ontology

⁶<https://github.com/polifonia-project/polifoniacq-dataset>

modules shaping the architecture of the Polifonia Ontology Network.

To streamline this process, we analysed the CQ embedding space generated and projected by IDEA. This is done by computing the sentence-level embeddings (a feature vector of fixed size) for each CQ in the PolifoniaCQ dataset. The latter can be considered as a point in a high dimensional space – providing a numerical summary of the question’s meaning [68]. Embeddings are computed via Sentence-BERT [333] due to its state of the art performance on a number of question-related tasks, including multi-lingual search and paraphrase detection.

An interactive visualisation of the PolifoniaCQ embeddings is available from a live Tensorboard Projector [366] which is set up and synchronised via IDEA⁷. The qualitative analysis of the embedding space, in addition to density-based clustering analysis under various parametrisations, have jointly facilitated the identification of common requirements (as nested clusters) and enabled the interactive exploration of the PolifoniaCQ dataset via similarity (c.f. Figure 3.2).

Matching CQs to ODPs. For each module/ontology, an XD iteration starts from selecting a coherent set of CQs. To address those requirements, existing solutions (ODPs) from ontologies or online catalogues of patterns are considered for reuse, extension, and specialisation. For instance, a CQ such as “Where and when a situation took place?” can be matched to the *TimeIndexedSituation*⁸ ODP, which represents temporal situations.

Here, IDEA supports the identification of “the CQ set” via the multi-lingual search feature. For example, an ontology designer looking for CQs related to places may express a search query as shown below in Listing 3.1.

Listing 3.1: *Search results for query “questions related to places” with similarity score.*

1	0.377	Where were the places in which musicians played?
2	0.368	Which are all organs near to geographic coordinates x, y?
3	0.341	What are geographically distinct features of organs from a region?
4	0.287	Where is the church/bell tower?
5	0.285	What is the provenance of the event attendees?
6	0.275	Which tunes which share melodic patterns or geographical origin?
7	0.265	What places did a musician visited in her career?
8	0.263	Where is the Bell Tower?
9	0.246	Where was a musical composition performed?
10	0.238	In which buildings was a musical composition performed?

⁷<https://polifonia-project.github.io/idea/category/competency-questions>

⁸<http://ontologydesignpatterns.org/wiki/Submissions:TimeIndexedSituation>

3.4. The Polifonia Ontology Network (PON)

Direct/indirect ontology reuse. Depending on the project’s requirements, reuse of ontologies and ODPs is direct and/or indirect [53]. The former approach directly includes/imports ontologies or part of them (e.g. individual entities, relations) thus introducing a dependency to any possible changes and availability. In indirect reuse, relevant entities and patterns from other ontologies are used as templates (replicated and extended) while being aligned to ensure interoperability. In Polifonia, we follow a hybrid approach: ArCo ontology [54] is directly reused since its development and maintenance involves one of the project’s partners (MiC), while others (such as DOREMUS) are indirectly reused and aligned.

Validation and testing. Ontology modules have been developed in close collaboration with domain experts and pilot leaders throughout the whole development cycle. This has allowed the ontology design team to leverage the domain expertise in Polifonia to technically validate our modules at different stages: from the collection and analysis of requirements, to iterations of ontology designs. Validation was facilitated by IDEA (at the CQ-level), and, at the modelling level, by the Graphical Framework For OWL Ontologies (Graffoo) notation [132] – providing a powerful visual language for coproduction activities. This has also been achieved through data snippets provided by the pilots, which have been modelled by our ontologies and triggered further iterations of improvements.

Overall, the involvement of domain experts from different institutions and background (complementary views and notions), the 10 pilots in the Polifonia project (reasonable diversity of application domains), and the use of collaborative workflows have also contributed to mitigate bias in the development of PON.

3.4 The Polifonia Ontology Network (PON)

The Polifonia Ontology Network (PON) provides a modular backbone of music ontologies to address both cultural heritage and more general queries in the music domain. As illustrated in Figure 3.3, PON v1.0 comprises 15 ontology modules that are organised thematically (colours, horizontal view) and hierarchically, to highlight their dependencies (vertical view). At the bottom of the architecture lies our **Core** module (providing general-purpose elements of design, ODPs, and alignments) and the reused ontologies. Four foundational models provide interoperability across PON through their abstract design: **Source**, **Instrument**, **Music Meta**, and **Music Representation**. These are specialised and extended in the

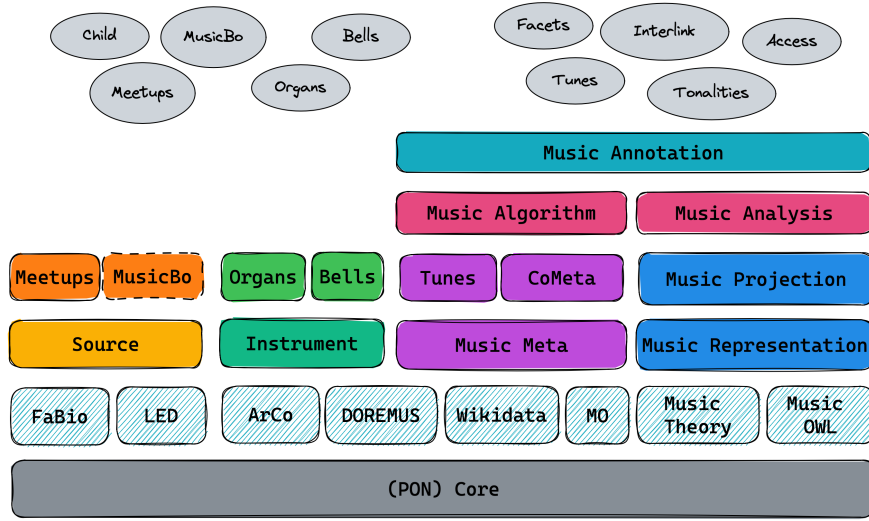


Figure 3.3: Overview of the main modules in the Polifonia Ontology Network, with Polifonia’s pilots as early adopters (grey circles). Foundational models (*Source*, *Instrument*, *Music Meta*, *Music Representation*) provide the backbone of PON, built on top of the *Core* module while leveraging the main ontologies reused directly or indirectly.

upper levels to add functionalities and contextualise specific domains.

A summary of PON modules is given in Table 3.2, with links to the repositories storing the modules with documentation, diagrams, and examples. Through our foundational models, PON ontologies can be applied to a wide set of music projects, and the modular design simplifies extensibility and maintenance. To facilitate this process, further documentation and tutorials are also being made available at <https://polifonia-project.github.io/ontology-network/>. An example of use involving 5 PON modules (besides *Core*) is shown in Figure 3.4.

3.4.1 Foundational models and their extensions and specialisations

The Music Meta module provides a rich and flexible ontology to describe music metadata related to artists, compositions, performances, recordings, broadcasts, and links. Music Meta focuses on provenance and interoperability – essential requirements for the integration of music datasets, which is currently hampered by the specificity of existent ontologies. The model is based on the Information-Realisation ODP [150], allowing to reduce the complexity of FRBR-based models, whose application in the music domain has raised concerns [335]. Given the relevance of this module, we will explore it in greater detail in Section 3.5.

3.4. The Polifonia Ontology Network (PON)

Module	Prefix	Outline	Repository
Core	core:	Elements of general reuse and ontology design patterns	/core-ontology
Music Meta	mm:	Achieving interoperability of music metadata	/music-meta-ontology
Music Representation	mr:	Foundational model to describe arbitrary musical content	/music-representation-ontology
Music Instrument	mop:	Instruments and their evolution through time and space	/music-instrument-ontology
Source	src:	Musical sources and their context of production	/source-ontology
Tunes	tunes:	A specialisation of Music Meta for folk music	/tunes-ontology
CoMeta	com:	An extension of Music Meta to represent music corpora	/cometa-ontology
Music Projection	mp:	Achieving interoperability of music notation systems	/music-projection-ontology
Organs	organ:	A rich descriptive model of organs and building methods	/organs-ontology
Bells	bell:	Describing bells, bell towers and bell ringers	/bell-ontology
Music Algorithm	mx:	Computational methods for music and their parametrisation	/music-algorithm-ontology
Music Analysis	ma:	Music analysis through reasoning using modal-tonal theories	/music-analysis-ontology
Music Annotation	ann:	A wrapper of ontologies for music annotations (audio, symbolic)	/music-annotation-ontology
PON (full)	pon:	The whole Polifonia Ontology Network (imports all modules).	/ontology-network

Table 3.2: Overview of the modules in the Polifonia Ontology Network. All URIs are also accessible from <https://github.com/polifonia-project/ontology-network>.

The Tunes module extends and specialises Music Meta for folk music. The main novelty consists in grouping and describing *tunes* into “*tune families*” depending on their melodic similarity (an association requiring rich provenance description of the musicological analysis on the source); which also extends to lyrics families.

CoMeta reuses and extends Music Meta to describe arbitrary music collections, corpora, and datasets. Here, metadata is described at the *collection-level* (data curator, annotations provided, availability of audio music, etc.), and at the *content-level*, (e.g., the title, artist, release of each piece in a dataset). The design of CoMeta is informed by a survey of Music Information Retrieval datasets [271].

The Music Representation module provides a comprehensive schema to describe the analysis of musical objects (a score, an audio track, etc.) interpreted according to a theory. Fragments of a musical object (elements of a music object whose temporal location is uniquely identifiable) are described by annotations provided by an agent (e.g. expert annotator, algorithm). An annotation is either the subjective result of an analysis (e.g. a chord played in a specific section) or objective in nature (e.g. a note in a digital score). Each annotation describes some music content (e.g. notes, chords, etc.), which we refer to as a *musical projection* [256]. Annotations are formalised via our Music Annotation Pattern [89], whereas the definition of music projections is delegated to the **Music Projection** module. The generality of the module and its abstraction over the represented content enables the interoperability of different music annotation schemas. The module is aligned to MusicOWL [213], Music Notation Ontology [62], Music Note

Ontology [315], and our JAMS ontology (c.f. Section 3.4.2).

The Music Projection module formalises musical entities that can be subject of an annotation. This ranges from traditional musical notation (e.g. note, chords) to informal annotations (e.g. mood, danceability). The module is aligned with MusicOWL, Music Notation Ontology, Music Note Ontology, Music Theory Ontology [332], Chord Ontology [135], and Roman Chord Ontology⁹. This allows to integrate existing domain ontologies. Notably, we also harmonise different chord representations (Chord Ontology, the Roman Chord Ontology and the Tonality Ontology) based on the Unified Model of Chords in Western Harmony [187].

The Instrument Module describes musical instruments as mediums of performance and their technical properties. Given that numerous taxonomies of instruments into *groups* and *families* exist (e.g. Hornbostel-Sachs, MIMO, MusicBrainz) and finding common categorisations is an open problem [224], our module provides an abstraction capable to express arbitrary classifications. This is achieved by leveraging the Information-Realisation and the Collection ODPs. Overall, the module allows to: (i) refer to instruments as entities (an instrumentation of a piece for “piano” and “viola”) as well as conceptually (e.g. a viola has 4 strings); (ii) support the integration with different taxonomies and vocabularies, such as [248]; (iii) describe the evolution of instruments in time and space (e.g. a viola as a cultural heritage object being relocated). This provides a foundational level where contributors can “plug” their instrument-specific ontologies [422].

The Bells module extends **Instrument** to describe bells by means of measurable, intrinsic aspects such as weight, materials, conservation status. The main entities contextualising bells are: (i) the author(s), such as the foundry who built the bell; (ii) the agencies that played some role e.g. the agency that took care of cataloguing the bell; (iii) the place(s) where it has been located; (iv) the tower(s) where the bell has been included; (v) the tools that the set of bells is played with; (vi) documents related to the bells, e.g. bibliographies, protective measures.

The Organs module extends **Instrument** to describe organs as (i) a musical instrument consisting of parts; and (ii) as a focal point of *projects* detailing its changes throughout time. To address the former, we used the Parthood pattern from the DOLCE ontology¹⁰. The entities of the ODP, **Whole** and **Part** make

⁹<https://github.com/polifonia-project/roman-chord-ontology>

¹⁰<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl>

3.4. The Polifonia Ontology Network (PON)

possible the specification of the whole instrument and its parts. In the ontology, the **Whole** entity refers to the organ instrument, and the **Part** entity refers to the parts of the organ that are **Console**, **WindSystem**, **Case**, **Division**, and **Action**.

The Source module represents various sources of music-related information. These include manuscripts, textbooks, articles, interviews, reviews, comments, memoirs, etc. of different scope and format (physical, digital). The module aims to provide general support to describe information related to the *creator* and *type* of the source, the *time* and *place* when/where it was created, the *context of production* and *usage*, and the *subject* and *goals*. Although this conceptualisation leans towards bibliographical sources, the module provides expressivity to indicate multimedia documents (e.g. images of scores, audio recording, video). For example, a video recording of a performance can be considered as a musical source – providing documentary evidence of a composition e.g. during an event.

The Meetups module describes encounters between people in the musical world in Europe from c. 1800 to c. 1945. Historical meetups, which are the main subject of this module, are described by means of four main components: the people involved in the meetup, for instance, the person that is the subject of interest and the people interacting in the event, the place where the encounter took place (e.g., city, country, venue), the type of event, the reason (e.g., music making, personal life, business, among others) and the date when it took place.

The MusicBO module is developed by following a *KG-to-ontology* process [269]. Ontological axioms, grouped into *patterns*, are empirically generated from the MusicBO knowledge graph – which is built from a textual corpus on music performances and encounters between music-related agents in Bologna since the 17th century. Such patterns include information about the probability of axioms to *happen* (as they are derived from the data). For instance, the probability of instances of the pattern *compose* situation (the process of creating art) to have **NaturalPerson** as range of the **artist** property, is higher than the probability of having an **Organisation** as a composer. In sum, the content of the ontology module is highly dependent on the KG, and the most populated and described entities are: persons, places, organisations, works of art, theatres, and books.

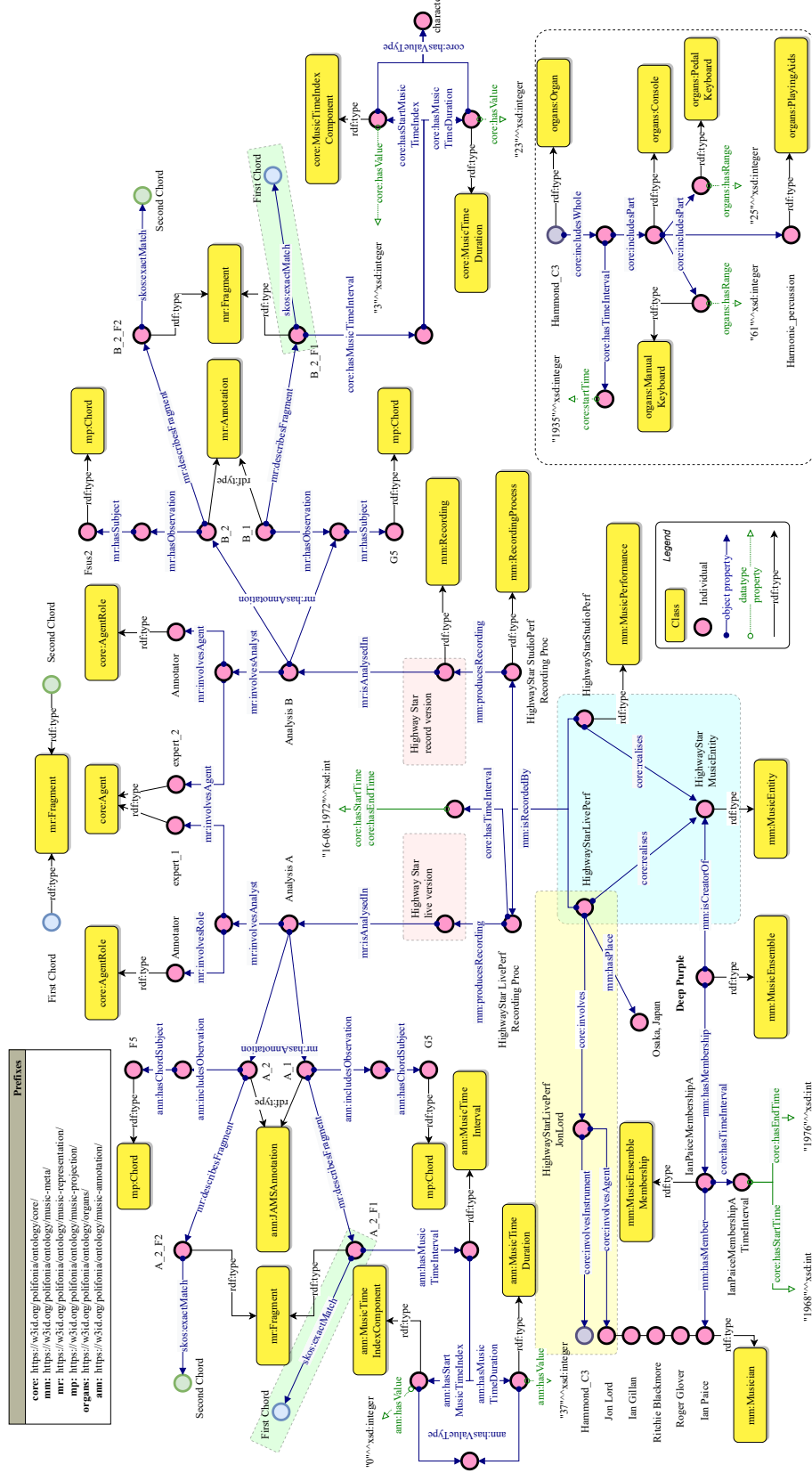


Figure 3.4: Graffoo [132] example of “Highway Star” by Deep Purple using 5 PON modules to describe: metadata information (Music Meta, bottom), instrument (Organs, bottom-right) and annotation of musical content on two audio recordings via the Music Representation, Projection, and Annotation modules, either related to a studio (top-right) or a live (top-left) performance of the same piece. We remark how the two musical annotations are made interoperable via PON despite their profound differences (JAMS [203] and text, respectively) as they refer to the same fragment.

3.4.2 Modules for analysis and annotation of music

The Music Algorithm module formalises algorithms that can operate on music metadata (using the Meta module), and musical content (via the representation module). The module commitments are similar to those defined by Diamantini et al. in [117]. Indeed, an algorithm is characterised by three main components: a *formalisation*, which can be theoretical (e.g. pseudocode) or executable (e.g. using a programming language); a *parametrisation* (e.g. input data); and the kind of *task* it solves. The latter defines a set of entities that are processed alongside the input and output data requirements and the final goal achieved. The module allows theoretical and quantitative performances to be represented in the context of the algorithm’s parametrisation. Through an abstract and general definition, the formalisation in Music Algorithm can be seen as a general pattern, capable of representing any algorithm regardless of the domain of application. In the context of music, the output of the algorithm is considered an analysis, which is then represented via the **Representation** module.

The Music Analysis module allows for the analysis of musical pieces using historical and present-day established musical theories: the *modal* and *tonal* theories. Through the use of this framework, different subjective analyses can be unified – overcoming the limitations imposed by a “global” theoretical perspective. Different theoretical viewpoints can be used for the interpretation of the same piece. Currently, two historical theories are implemented: Zarlino (1558) and Praetorius (1619) [168, 169]. Through the use of formal reasoning and a comprehensive axiomatisation, the ontology is able to automatically infer the theoretical interpretations of a musical piece and its evolution in time and space.

The Music Annotation module provides different music annotation models to accommodate musicological and information retrieval use cases. The primary objective of this module is to enhance support for other descriptive systems, thus increasing interoperability and conversion possibilities from various music annotation formats. Indeed, all our models are logically interconnected through **Music Representation**. A fully fledged annotation model here is the JAMS Ontology [90]¹¹, which is detailed in Dataset 4 (c.f. Section 4.3.1). This ontology mimics the structure of a JAMS (JSON Annotated Music Specification for Reproducible MIR Research) document [203]. It semantically describes and connects

¹¹<https://github.com/polifonia-project/jams-ontology>

all the elements of the JAMS specification (`Annotation`, `Observation`, etc.), including the music metadata and the annotation contents using the `Music Meta` and `Representation` modules, respectively.

3.5 The Music Meta Ontology

Music Meta is part of Polifonia Ontology Network (PON), from which it imports the CORE module (c.f. Section 3.4). The ontology (prefixed as `mm`) is available at the following URI: <https://w3id.org/polifonia/ontology/music-meta/>, and is released as open source project under the CC-BY 4.0 on GitHub¹².

From FRBR to Information Objects/Realisations

At the core of Music Meta lies the use of the Information-Realisation (IR) ODP [150]. An *information object* is a non-physical social object carrying information that can have one or multiple materialisations (*information realisations*). Each realisation is a particular physical object, or event, realising the *information object*, or involving the latter as a participant. Both information object and realisation are intended as Information Entities (IEs), i.e. (social) objects created and/or used to communicate, reason, and specify new entities. This allows to distinguish between a piece of information (e.g. the *content* of a composition) from how it is materialised (e.g. as a performance).

On the other hand, both the Music Ontology [324] and DOREMUS [64] are built on top of different flavours of FRBR¹³ (FRBRer and FRBRoo, respectively). FRBR is a conceptual model describing bibliographic resources at four levels: *Work*, *Expression*, *Manifestation*, and *Item*. In contrast, the two levels of the IR pattern map to *Expression* and *Item*, since *Work* and *Manifestation* are said to provide non-informative conceptualisations [150]. Moreover, [335] argues that FRBR's Works – intended as “entities that pre-exist expressions”, cannot represent improvisations or traditional music, as they do not derive from a formal composition process leading to a realisation. FRBR's Work is often ambiguously intended as an entity retrospectively created for grouping multiple expressions for cataloguing needs. As for the Manifestation level, while its representation is straightforward in the bibliographic domain (e.g. the printed version of a book), its correspondence in the music domain is not fully intuitive, as it may relate to

¹²<https://github.com/polifonia-project/music-meta-ontology>

¹³<https://www.ifla.org>

specialisation of persons who can optionally be associated to a medium of performance (e.g. voice, guitar), and be part of a music ensemble (e.g. `MusicGroup`, `Orchestra`, `Choir`). Depending on the data available, the latter can be expressed either through a membership relationship (`core:isMemberOf`), a specialisation of the former, such as `mm:isSingerOf`, or through a `mm:MusicEnsembleMembership` when the period of participation of the musician is available.

All music artists can be associated to (one or more) `mm:MusicGenre(s)`, express influences or collaborations, and share a period of activity. Here, the start date refers to the foundation for music ensembles, whereas the end date is used for discontinued projects for algorithms.

Music inception

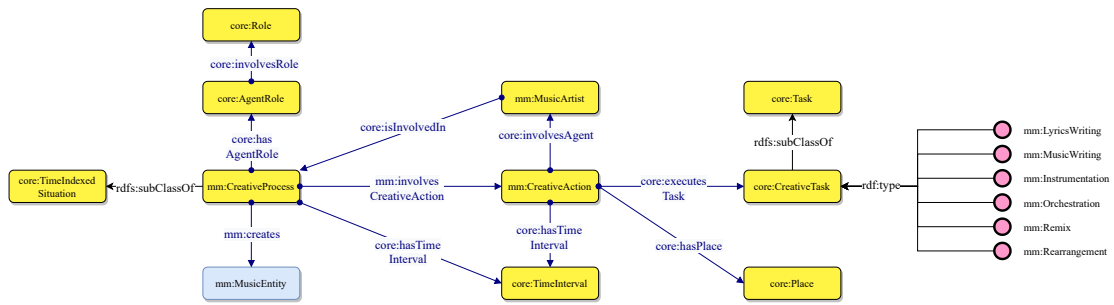


Figure 3.6: *Abstracting music inception as an product of a creative process, involving music artists in activities (music writing, instrumentation, etc.), defined in time and space and according to different roles.*

The focal point of Music Meta is the `mm:MusicEntity` class (Figures 3.6 and 3.7). This class represents an Information Object, which is defined as the sum of all the elements that make up a piece of music. A Music Entity is composed of several components, including lyrics (generalised through `mm:Text` to also account for `mm:Libretto`), the entailed musical content (`mm:AbstractScore`) and its instrumentation (`mm:Instrumentation`).

A `mm:AbstractScore` provides an abstraction to describe the musical properties of an entity, such as the form of a piece (`mm:FormType`), its constituent parts (e.g. `mm:Movement` or `mm:Section`), and its key (`mm:Key`). Datatype properties also describe the composition tempo (`mm:tempo`) and its order (`mm:orderNumber`). A `mm:Instrumentation` can instead be formalised in a `mm:Score`, which can be either digital or paper. Through the score, the instrumentation describes one or more `mm:MediumOfPerformance`, each of which has a cardinality (e.g. 3 violins).

In sum, the model provides flexibility across periods and genres as the proposed classes allow generalisations to be made about the text, the musical composition and its arrangement. Through the specialisation of classes, depending on the target domain/application, specificity can easily be achieved (c.f. Section 3.1.1). For example, a tune family can be seen as a `mm:Collection` encompassing several tunes (as music entities) based on specific criteria (e.g. similarity, provenance).

Figure 3.7: Describing a music entity and the elements it contains: *Text*, *AbstractScore* and *Instrumentation*.

The realisation of a `mm:MusicEntity` is exemplified by `mm:MusicalPerformance`, which can be either live or in a studio. As illustrated in Figure 3.8, the place and time intervals are described by `core:Place` and `core:TimeInterval` classes – in-

Chapter 3. Representing Musical Knowledge

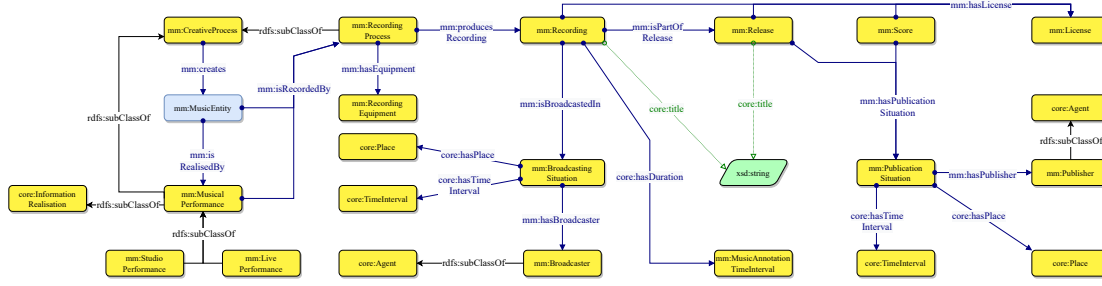


Figure 3.8: Describing performance, recording, broadcasting, publication, and licensing.

volving one or more music artists (optionally, with a specific role). A performance may also create a new `mm:MusicEntity` if, e.g., the execution differs significantly from the original version.

A Music Entity can also be recorded by means of a `mm:RecordingProcess`, which is a subclass of a `mm:CreativeProcess`. This makes it possible to describe information about both the production (e.g., producers) and the technical aspects of it (e.g., sound engineer, equipment used). The recording process produces a `mm:Recording`, which is contained in a `mm:Release`.

Information about the broadcasting of a recording is modelled through the `mm:BroadcastingSituation` class (an instance of the Situation ODP [148]), which describes when and where the song was broadcast, and by which broadcaster (`mm:Broadcaster`).

Publishing and licensing information

The `mm:PublicationSituation` class describes information about the publication of a release, which is common to the publication of a `mm:Score` (c.f. Figure 3.8). For both a release and a score, it describes when and where they were published, and by a `mm:Publisher`.

Licence information is described by the `mm:License` class, which applies to records, releases and scores.

Modelling links and integrating provenance

We propose a pattern based on *RDF** [182] to describe the provenance at different levels (Figure 3.9). The use of *RDF** is particularly useful for this purpose, as it allows to embed provenance information to every triple in the dataset. This simplifies and streamlines the model, eliminating the need for n-ary relations or

reification for each triple.

The proposed pattern is straightforward and comprises the `core:Reference` class, which describes the source of the reference (using the class `core:Source`) and the method used to obtain the annotation (using `core:SourceMethod`). Additionally, the datatype properties `core:confidence` and `core:retrievedOn` describe the confidence of the annotation and the date it was produced, respectively.

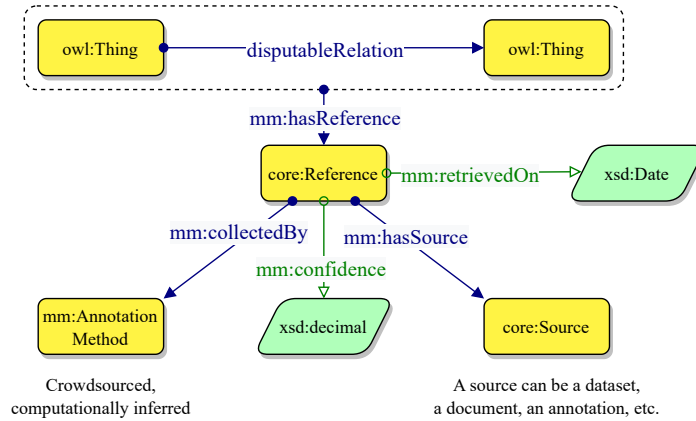


Figure 3.9: Our pattern to describe provenance with *RDF**.

3.5.2 Conversion rules and code support

To facilitate the reuse of Music Meta and its data conversion into OWL/RDF Knowledge Graphs, we developed `PyMusicMeta` – a library to map arbitrary music metadata into RDF triples. This enables a practical and scalable workflows for data lifting to create Music KGs without expert knowledge of our ontological model. The library is developed in Python as an extension of `RDF-Lib` [40]. With each triple, `PyMusicMeta` adds alignments to the supported schema whenever possible. For example, the pseudo triple `<DavidBowieURI, rdf:type, mm:Musician>` in Music Meta will be complemented with:

- `<DavidBowieURI, rdf:type, http://purl.org/ontology/mo/MusicArtist>` for Music Ontology;
- `<DavidBowieURI, rdf:type, http://erlangen-crm.org/E21_Person>` for DOREMUS (via the Erlangen Conceptual Reference Model [270]); and
- `<DavidBowieURI, rdf:type, https://www.wikidata.org/wiki/Q639669>` for Wikidata;

to achieve interoperability of the Music KG.

3.6 Adoption and impact

We provide evidence of PON use by Polifonia pilots (Interlink, Tonalities, Meetups, Bells, and MusicBO), which have contributed 6 musical heritage KGs (Section 3.6.1); potential interest of reuse and opportunities for the Semantic Web and Music Technology communities collected from an online survey (Section 3.6.2); early adopters and ongoing synergies from the Polifonia Stakeholder Network for PON validation and annotation of cultural and industrial datasets (Section 3.6.3).

3.6.1 Current use by Polifonia pilots

Interlink has released ChoCo and Harmory KGs. Choco [90] provides 20K+ harmonic annotations of scores and tracks, that were integrated from 18 chord datasets¹⁵. The KG uses the JAMS ontology in **Music Annotation**, and the **Roman** ontology from the **Music Projection** module. Harmory [92] is a KG of interconnected harmonic patterns derived from ChoCo, and aimed at human-machine creativity (pattern discovery, chord generation, harmonic similarity).

Tonalities KG includes data¹⁶ from 377 MEI scores and their annotations w.r.t. theoretical concepts (roots, harmonic progressions, dissonant patterns, cadences, etc.), using the 2 theoretical models in the **Music Analysis** module.

Meetups KG describes 74K+ historical meetups from c.1800 to 1945, mentioning 51K+ people from 5K+ places in Europe¹⁷. It uses the **Meetups** ontology and is extracted from 1K artists' biographies on open-access digital sources.

Bells KG describes 88 bells catalogued by the Italian Ministry of Culture¹⁸. It relies on the **Bells** module and is part of the ArCo KG – the largest Italian cultural heritage KG from the Italian General Catalogue of Cultural Heritage.

MusicBO KG is built via text-to-KG methods [269] on a collection of 137 documents¹⁹ on performances and encounters between musicians, composers, and critics happened in Bologna from the 17th century. As mentioned in

¹⁵<https://polifonia.disi.unibo.it/choco/>

¹⁶<https://data-iremus.huma-num.fr/sparql>

¹⁷<http://data.open.ac.uk/context/meetups>

¹⁸<https://dati.cultura.gov.it/sparql>

¹⁹<https://doi.org/10.5281/zenodo.6672165>

Section 3.4, the KG²⁰ is used as input to the *bottom-up* modelling of the MusicBO ontology.

3.6.2 Survey of interest for future applications

To gather interest of adoption, we conducted an online survey in which we ask potential adopters 14 questions regarding their background, relevance, and interest in using music ontologies. The survey was conducted via Google Forms, and distributed in the Semantic Web (SW), MIR, and Digital Humanities mailing lists – gathering a total of $N = 61$ responses. Among our respondents, 25 work in SW, 23 in MIR, 26 in Musicology. Most of them have encountered the need for modelling music related data and resources with ontologies (65.6%), focusing primarily on music metadata (45) theory and notation (29), annotations (25) and instruments (28); with 75% doing research or project work related to music data with multiple stakeholders.

Participants were asked to quantify the agreement with statements from 1 (absolutely disagree) to 5 (absolutely agree), 3 being a neutral response (NR; neither agree nor disagree). Results are illustrated in Figure 3.10. From questions 6-14 we found that: 49.2% find the reuse of existing music ontologies to be challenging (with 42.6% NR), and the same can be said about the interoperability of existing ontologies (57%; 36% NR), their lack of coverage of concepts related to music history and music cultural heritage (57.3%; 32.8% NR); and the lack of large datasets of competency questions for this domain (63%; 34% NR). We also find strong evidence for potential reuse of PON, as participants would be interested in using ontologies for music metadata (78.7%), sources (80.3%), musical instruments (70.5%), and music content (57.4%; 21.3% NR), as well as a CQ dataset for musical heritage (65%; 26.7% NR).

²⁰<https://polifonia.disi.unibo.it/musicbo/>

Chapter 3. Representing Musical Knowledge



Figure 3.10: Selection of questions 4, 6, 7-14 from the online survey, where responses are expressed on a Likert scale ranging from 1 (strongly disagree) to 5 (strongly agree).

3.6.3 Adoption by Polifonia Stakeholders

In addition to internal and potential adopters, industrial and institutional stakeholders in the Polifonia Stakeholder Network have also expressed interest to use PON resources. These include the **Digital Music Observatory**, concerning the use of **Music Meta** and **Source** to annotate the numerous music resources of the consortium; and the **Université Catholique de Louvain** where Anne-Emmanuelle Ceulemans uses the **Music Anlysis** module for studying the annotation of cadences in Josquin des Prez (composer of High Renaissance music).

We have also planned work with **Deezer**, **Songfacts**, and **MusicID** for the evaluation, extension, and reuse of **Music Meta** driven by their resources; and collaborations with the EU H2020 **MuseIT**²¹ project to extend the ChoCo KG.

3.6.4 Availability, sustainability, and FAIRness

PON namespaces are introduced in Section 3.4, and permanent Uniform Resource Identifiers (URIs) were created with the W3C Permanent Identifier Community Group. PON is under version control on public GitHub repositories (c.f. Table 3.2), and all repositories are also published on Zenodo (with associated DOIs) under the CC-BY 4.0 licence. The storage of all resources on GitHub guarantees their persistence beyond the project, with the University of Bologna and the Italian Ministry of Culture (MiC) committed to host and maintain PON on the long term. We also remark that PON is reused as a sibling ontology project of ArCo by MiC [54].

3.7 Conclusions

This chapter presented the creation and development of the Polifonia Ontology Network (PON), a collection of expressive ontologies aimed at addressing the challenges of interoperability in musical cultural heritage. Through the Polifonia project, we applied and extended the XD methodology for ontology engineering, incorporating multidisciplinary and domain-specific requirements. Our methodological innovations, such as the IDEA framework for NLP-assisted ontology co-design, helped ensure that the design of PON was grounded in real-world needs and technological advancements.

²¹<https://www.muse-it.eu/>

Chapter 3. Representing Musical Knowledge

PON (v1.0) comprises 15 new ontologies, along with the release of the PolifoniaCQ dataset containing 361 competency questions, all made available under an open license (CC-BY 4.0). Furthermore, we provide evidence of current and potential reuse by institutional and industrial stakeholders, demonstrating the practical relevance and impact of PON.

As a next step, we plan to perform an extensive competency question-driven evaluation of PON's modules to further refine the ontologies. Additionally, we will continue to support stakeholders and early adopters in reusing, extending, and maintaining the ontologies and knowledge graphs over the long term. This includes ongoing work to specialise the Music Meta model for the integration and release of new music knowledge graphs, both in the cultural heritage and music industry domains. Furthermore, we plan to extend PON modules by incorporating novel music theories, for instance through the Music Analysis Module, to broaden the scope and applicability of the ontology.

Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

4.1 Introduction

As discussed in Chapter 3, music data is fragmented across numerous datasets, each using its own conventions, tailored to address specific tasks or applications. This fragmentation arises from representational issues in both metadata and music content, addressed in RQ1 (Section 1.1.1). While the metadata representation often lacks standardization and interoperability, making it difficult to integrate different datasets, the content representation is scattered across a multitude of formats and notational systems (c.f. Section 2.3).

This fragmentation is well exemplified by datasets containing harmonic data. As discussed in Chapter 2.1.3, harmony is a widely studied dimension in music theory [312, 351] and music analysis [188]; where functional harmony provides a set of rules for moving to and from the *tonic* – the most stable note in a piece, allowing to relate chords to each other, and to the main harmony.

Chords are the basic constituents of harmony, which jointly define the harmonic

structure of a piece. Individually, a chord is defined as a simultaneous occurrence of several music sounds, producing harmony [165]. Depending on the notational system and the annotation conventions, a chord can be associated with a name, or label. For example, the chord **G7** (typically read as “*G dominant seventh*”) in the key of C major, contains the notes $G - B - D - F$ and may create tension partly due to the tritone relation between B (leading tone) and F (the seventh of the chord). These intervals to the root characterise the intrinsic harmonic properties of chords, as well as the relationships with other chords in the same harmonic progression [32].

Perceptually, some chords sound more stable, final and resolved, while others sound unstable and tense – a phenomenon that is salient both to young children and to adults, even from diverse cultures. However, the definition of harmony differs vastly across time, genre, and individuals [225], reflecting a great heterogeneity in terms of harmony perception [189, 265]; and in this work, we focus on Western tonal music tradition. In this regard, harmony exerts an affective role: major harmonies tend to represent positive emotions (happiness, joy, triumph, etc.); minor triads express “negative” emotions (sadness, anger, etc.); diminished triads (chains of minor thirds) indicate suspense and other disorienting sentiments, while augmented triads (all major third intervals) tend to create senses of spookiness, extreme dark emotions, and mystery [74].

Computationally, the automatic analysis of chord progressions has addressed several tasks in information retrieval – from the detection of cadences, patterns, structures in music, to the introduction of harmonic similarity measures for cover song detection, symbolic search, and content-based music linking. Progress in machine learning research has also sparked interest in computational creativity applications, such as arrangement generation, continuation, infilling, and automatic music composition with harmonic conditioning [128] (e.g. generating melodies from a given harmonic template) to name a few.

To account for the evolution of harmony and explain its subjective and genre-specific differences, while enabling the aforementioned applications, the availability of large, diverse, and reliable chord data is fundamental. However, several different chord notations exist (Harte, Roman, ABC, Leadsheet, etc.), each with different levels of expressiveness, in a large number of disconnected chord datasets that are hard to combine [52]. This poses a challenge for combining existing chord datasets into larger ones. Existing approaches address this issue by focusing on

scale, and publishing large numbers of chord annotations. For example, UltimateGuitar¹ offers a collection of 1.1M+ songs annotated by a community of 12M+ musicians. Chordify² addresses the challenge of scalable chord annotation by applying methods for automated chord estimation. However, none of these approaches solves the problem of integrating chord datasets complying with the following desiderata: **(a)** high quality of the data; **(b)** precise timing information; **(c)** release through open licences; **(d)** use of different chord notations; **(e)** diversity of music genres; and **(f)** large scale. The problem is exacerbated by the little reuse of standard formats for music annotation. In the context of this thesis, *music annotation* is defined, in a broad sense, as the outcome of a music analysis carried out by a domain expert on the musical surface (a score, a recording) to identify and locate elements of interest (e.g. chords, segments, patterns, etc.), following an established methodology. For example, if the goal of a harmonic analysis is to identify chords from a composition, a music annotation may correspond to a list of chords together with a reference to their onset and offset (i.e. when they occur in the piece).

4.1.1 Our contribution

In this chapter, we introduce the *Chord Corpus (ChoCo)*, a comprehensive KG of harmonic annotations and a workflow designed to facilitate the development of musical harmony Knowledge Graphs leveraging PON (c.f. Section 3). These contributions, published in [90], directly address RQ2 (Section 1.1.2) by focusing on strategies for unifying symbolic music datasets to standardize diverse digital formats and annotation practices, thereby tackling the fragmentation challenges within existing chord datasets.

The workflow we present encompasses the curation, transformation, and integration of over 20,000 human-made, high-quality harmonic annotations from 18 highly heterogeneous chord datasets (desiderata *a*, *b*, *f*), following the JAMS data structure as annotation model. The resulting annotations are rich in provenance data (e.g. metadata of the annotated work, authors of annotations, identifiers, etc.) and refer to both symbolic music notation and audio recordings, while encompassing different notation systems (desideratum *d*). After semantically enriching, extending, and standardising these annotations under the JAMS definition,

¹<https://www.ultimate-guitar.com/>

²<https://chordify.net>

we use the PON ontologies described in Chapter 3 to release the *ChoCo Knowledge Graph* – providing fine-grained semantic descriptions of chords, opportunities for chord interoperability, and 4K+ links to external datasets. All data and code are released using open data licences (desideratum *c*). We also show evidence of interest and use of ChoCo, and postulate its value for the SW and MIR communities at enabling the study of harmony through large scale data.

Specifically, the main contributions are summarised as follows:

- A generalised data curation framework to semantically integrate MIR harmonic datasets and represent chords from a large variety of formats (JSON, CSV, LAB, TXT, SQL, MusicXML, iReal, mgu, sku, ABC, etc.) as JAMS annotations.
- A large dataset and KG standardising, enriching, and integrating 18 existing chord collections in the literature. ChoCo is released both as a JAMS dataset and an RDF Knowledge Graph, to accommodate the requirements and needs of different communities (MIR, Musicology, SW, etc.).
- Demonstrations of the utility of both the workflow and the resulting chord corpus. These examples illustrate its relevance to both MIR and SW fields, including the use of the workflow to describe other musical elements, such as melodic patterns [353].
- Evidence of potential adoption and community interest, by conducting a survey targeting potential users, asking ten questions related to the relevance of chord data in their work, and their interest in adopting the dataset and workflow.

ChoCo achieves interoperability of harmonic datasets at three levels: metadata, annotation format, and chord notation. The interoperability at metadata and annotation format levels is implemented by integrating metadata from different sources, at the parsing level, and by leveraging the JAMS annotation standard to store harmonic annotations, consistently. Chord notation interoperability is achieved by converting chords to three reference notational systems (desideratum *d*) – bridging them via the Harte notation [181]. The outcome of this approach enables the use of these integrated collections as if they belonged to the same dataset and underpins the automatic generation of Music Knowledge Graphs. In addition to the conversions, ChoCo provides the original annotations in each JAMS file, along with rich provenance descriptions that keep track of the original sources.

4.1.2 Chapter Structure

This chapter is organized as follows: we begin with a review of the *related work* in Section 4.2, providing context for the challenges and existing approaches to harmonizing and integrating chord datasets in MIR.

The development of *ChoCo* is detailed in Section 4.3, where we present the methodology used for creating the dataset. This includes an overview of the data incorporated into ChoCo, the process of converting the data into a unified format, the conversion of chord annotations, and the creation of the resulting KG.

We then provide *descriptive statistics* in Section 4.3.2, offering an overview of ChoCo data at two different levels: the metadata associated with the music tracks, including their identifiers and links, and the content of the music annotations.

In Section 4.4, we present the *technical validation* of the dataset, demonstrating the reliability and completeness of ChoCo.

The chapter continues with *usage notes* in Section 4.5, where we discuss how both the dataset and the workflow have been employed, along with potential future applications.

Data availability and licensing information are outlined in Section 4.6, providing details on how ChoCo can be accessed and used under open licenses.

Finally, we present the *conclusions* in Section 4.7, summarizing the main contributions and future directions for the workflow and its applications.

4.2 Related Work

In the last decade, numerous systems and formats have been proposed for representing and storing musical annotations [187]. Some have been more successful than others, but no system has prevailed as a reference standard. Some systems are focused on symbolic music and are domain-specific (e.g. DCMLab, Roman-Text for harmonic analyses), embed annotations in the score (MusicXML, ABC, etc.), or propose variations of tabular formats to account for audio and symbolic music (LAB and xLAB). In the audio domain, JAMS (JSON Annotated Music Specification) [203] has emerged as a system to uniformly represent music annotations of different types and granularity, that is efficiently built on top of the JSON serialisation standard. JAMS is also supported by software libraries for dataset manipulation [34] and for the evaluation of MIR methods [323].

However, combined efforts of MIR and SW researchers to address (chord) anno-

tation data interoperability have been scarce. While MIR has contributed a great deal of music datasets, predominantly containing music annotations to train and evaluate computational methods for music analysis, SW technologies and principles can easily address the data integration problem at scale [52]. Nevertheless, the scarcity of semantic models for music annotations has hampered this vision, and more research efforts are hence necessary to devise domain-specific ontologies that can efficaciously address the interoperability issue through reuse and alignment. In addition, this kind of musical knowledge is also underrepresented in Knowledge Graphs [194], which are usually built from other knowledge archetypes such as logic statements or textual corpora. The lack of musical knowledge in the Semantic Web also limits our understanding of knowledge expressed in modalities other than text (e.g. images, music) and its challenges: semantic relations that have not been formalised yet, integration of multimodal datasets, etc.

Specifically for harmonic data, various chord collections have been published (see Table 4.1) making harmony annotations available, albeit through highly heterogeneous and non-interoperable notations (Harte, Leadsheet, Roman, ABC) and formats (JAMS, JSON, MusicXML, LAB, etc.). Other databases, such as UltimateGuitar and Chordify [98], focus on automation and scalability. These are achieved by annotating millions of songs via crowdsourcing or chord recognition algorithms, but have an inherent cost in annotation quality. Therefore, none of these approaches solves the problem of semantically integrating chord annotation datasets while meeting all the aforementioned desiderata (*a-f*).

The challenge of supporting interoperability of music content-related data has been the subject of relevant efforts in the last decade, especially supporting their evolution, reuse, and sustainability [401, 160, 209] according to FAIR data principles [406] and through Semantic Web technologies.

In Section 3.2, it is possible to find a comprehensive list of ontologies specifically developed for describing musical data.

Some of these ontologies are the backbone of large music notation knowledge graphs. For example, the MIDI Ontology [272] has been used to generate the MIDI Linked Data Cloud³, a large knowledge graph interconnecting 300K+ MIDI files through 10B+ triples of music-related linked data addressing music content rather than metadata. This misses, however, explicit chord information that could be useful for the symbolic analysis of harmony. MusicOWL [213] has

³<https://midi-ld.github.io/>

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

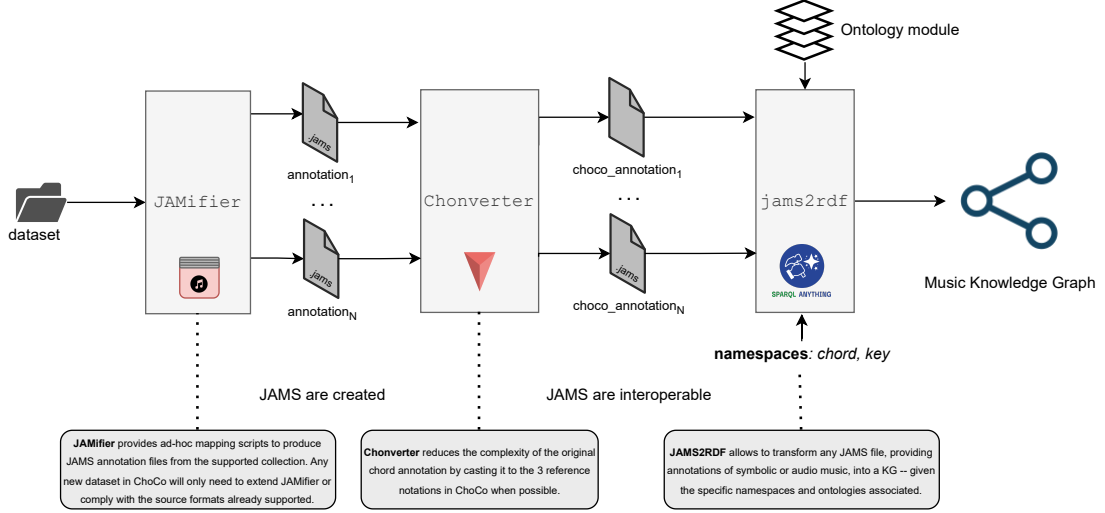


Figure 4.1: Overview of our data transformation workflow, generalised for arbitrary music annotations, and used here for chord and key annotations prior to the generation of the ChoCo Knowledge Graph. The *JAMifier* ingests chord collections (where metadata and music annotations follow collection-specific conventions and formats) to generate a JAMS dataset. This achieves two integration levels, as all metadata are consistently re-organised, and the music annotations (i.e. chord progressions, in this case) are all encoded and stored in separate JAMS files – one per track/score. The *Chonverter* achieves notational interoperability among collections by converting the original annotations to the same notational families. Finally, *jams2rdf* leverages notation-specific ontologies to generate RDF triples and create a Music Knowledge Graph.

been used for producing the Linked Music Score Dataset⁴ knowledge graph, representing elements of 43 historical scores from the Münster University Library. Yet, none of these previous efforts successfully addresses the challenges *a-f*); especially providing representations that meet the standards and the needs of different communities (e.g. JAMS for MIR, Musicology, and RDF for Semantic Web, Digital Humanities, etc).

4.3 ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

4.3.1 Methods

The general workflow to produce ChoCo is illustrated in Figure 4.1. We describe the resources contained in ChoCo (Section 4.3.1), and the data transformation

⁴<https://linkeddata.uni-muenster.de/datasets/opendata/ulb/musicscores/>

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

Collection	Type	Notation	Original format	Annotations	Genres	Ref
Isophonics	A	Harte	LAB	300	pop, rock	[258]
JAAH	A	Harte	JSON	113	jazz	[129]
Schubert-Winterreise	A, S	Harte	csv	25 (S), 25*9 (A)	classical	[402]
Billboard	A	Harte	LAB, txt	890 (740)	pop	[48]
Chordify	A	Harte	JAMS	50*4	pop	[225]
Robbie Williams	A	Harte	LAB, txt	61	pop	[116]
The Real Book	S	Harte	LAB	2486	jazz	[260]
Uspop 2002	A	Harte	LAB	195	pop	[28]
RWC-Pop	A	Harte	LAB	100	pop	[161]
Weimar Jazz Database	A	Leadsheet	SQL	456	jazz	[309]
Wikifonia	S	Leadsheet	mxl	6500+	various	-
iReal Pro	S	Leadsheet	iReal	2000+	various	-
Band-in-a-Box	S	Leadsheet	mgu, sku	5000+	various	[99]
When in Rome	S	Roman	RomanText	450	classical	[273]
Rock Corpus	S	Roman	har	200	rock	[97]
Mozart Piano Sonata	S	Roman	DCMLab	54 (18)	classical	[188]
Jazz Corpus	S	Hybrid	txt	76	jazz	[163]
Nottingham	S	ABC	ABC	1000+	folk	[292]

Table 4.1: Overview of the 18 chord datasets currently included in ChoCo. Letters “A” and “S” are used to denote audio and symbolic (or score) music subsets, respectively – from which harmonic annotations are collected.

workflow to: produce JAMS datasets (**Jamifier**, Section 4.3.1), integrate the different chord notations (**Chonverter**, Section 4.3.1), and create a music knowledge graph (Section 4.3.1).

Chordal data in ChoCo

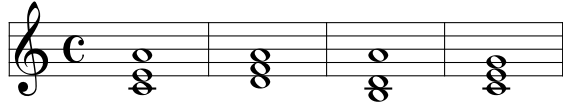
Table 4.1 summarises the source chord datasets (alias *subsets*, *collections*) that are integrated in our framework. ChoCo v1.0 integrates 18 high-quality chord datasets providing timed annotations of chord progressions in different formats (e.g. LAB, CSV, txt, mxl), notations (e.g. Harte, Leadsheet, Roman, ABC), and types (audio, symbolic). The rich and diversified nature of this resource, encompassing several genres/styles and periods, makes it the largest chord collection of its kind – with more than 20K annotated progressions. ChoCo’s collections can be categorised according to their generalised chord notation system: Harte, Polychord, Leadsheet, and Roman. An example of notation systems for the same chord progression is given in Figure 4.2.

Harte collections. Gather all collections with chords expressed in Harte notation [181]. The majority of these datasets are focused on pop/rock music, released in LAB format, and collected from audio music. Among them, *Isophonics* [258] provides chord, key, and structural annotations of a selection of albums by The

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

Beatles, Queen, Michael Jackson, and Carole King; *Billboard* [48] contributes similar annotations for a collection of songs sampled from the Billboard “Hot 100” chart in the United States between 1958 and 1991; *Chordify Annotator Subjectivity Dataset (CASD)* [225] augments a subset of Billboard with 4 expert annotators per song – to demonstrate the highly subjective nature of the chord identification/labelling task; *Robbie Williams* [116] contains key and chord annotations for 5 albums from this artist; *Uspop2002* [28] is a large scale dataset for music similarity, providing audio features, style tags, artist similarity ratings, as well as harmonic annotations for a smaller subset; *RWC-Pop* is a subset of the the Real World Computing (RWC) database [161], a cornerstone collection in MIR containing a great deal of instrumental and performance annotations, in addition to chordal information that was contributed by LabROSA. Among the other (non-pop) collections, we find the *Real Book* [260], providing chord annotations of several jazz standards from the homonymous book [383]; the *Audio-Aligned Jazz Harmony (JAAH)* dataset [129] contributing time-aligned harmony transcriptions from “The Smithsonian Collection of Classic Jazz” and “Jazz: The Smithsonian Anthology”; and finally, the *Schubert Winterreise* [402] multi-modal dataset, containing harmony and segment information of Franz Schubert’s song cycle “*Winterreise*” which were separately annotated from the score and from the audio (9 performances per score).

Leadsheet collections. Include four ChoCo collections using different flavours of the Leadsheet notation [136] for a variety of genres. These include the *Weimar Jazz Database* [309], providing rich cataloguing information, scores, YouTube links, and harmonic/melodic annotations of a selection of jazz solo transcriptions; *Wikifonia*, a copyright-free online publisher of sheet music in MusicXML format



Harte	A:min/b3	D:min	B:min7(*5)	C:maj
Polychord	C4,E4,A4	D4,F4,A4	B3,D4,A4	C4,E4,G4
Leadsheet	Am/C	Dm	Bmin7 no5	C
Roman [C major]	vi ⁶	ii	vii ^o 7[no5]	I

Figure 4.2: Example of a harmonic progression annotated using different notation systems, namely (i) Harte, (ii) Polychord (or decomposed chords), (iii) Leadsheet, and (iv) Roman Numerals.

which was discontinued in 2013; the *Band-in-a-Box (BiaB) Internet corpus* [99], containing human-generated chord annotations for BiaB – a commercial software⁵ that is used to generate accompaniment for musical practice; the *iReal pro* collection, a newly contributed chord dataset of various genres (jazz, blues, brazilian, latin, country, pop) that was created from the public playlists of iReal Pro⁶ – a commercial app with similar functionalities to BiaB.

Roman collections. Contain chord datasets providing harmonic annotations in Roman notation [9], and with more emphasis on classical music. A central dataset here is *When in Rome* [273], which already contains harmonic analyses from the *TAVERN* collection [111] (theme and variations for piano by Mozart and Beethoven), and the *BPS-FH* dataset [61] (Beethoven piano sonata); but also harmonic annotations from Monteverdi madrigals, Bach chorales and preludes, Haydn Op. 20 String Quartets, and a subset of nineteenth-century songs from the OpenScore Lieder corpus (Winterreise and Schwanengesang cycles from Schubert, Dichterliebe from Schumann, and several pieces by female composers). Notably, *When in Rome* is an actively maintained corpus where new harmonic annotations (in RomanText format) are also contributed and internally validated by experts. As a growing corpus of functional harmonic analyses, we plan to support the integration of future releases within ChoCo. Other Roman collections include the *Rock Corpus* [97], providing harmonic analyses, melodic transcriptions and lyrics information produced from a sample of Rolling Stone magazine’s list of the “500 Greatest Songs of All Time” in 2004 (pages 65 - 165); and *Mozart Piano Sonata* [188], featuring harmonic, phrase, and cadence analyses of all piano sonatas by Mozart.

Other collections. Include *Nottingham* [292], a dataset of British and American folk tunes, (hornpipe, jigs, etc.) released in ABC format; and the *Jazz Corpus* [163], providing harmonic analyses of jazz standards using both Harte-like and functional notations, the latter of which is akin, in purpose, to Roman numerals.

Chord datasets not included in ChoCo. Although other collections providing harmonic information exist in the literature, some of them were currently discarded for the reasons explained below. The *Leadsheet* dataset [417] separately annotates chord progressions for each segment (e.g. intro, chorus) but does not provide information on how structures are laid out in the piece. *GuitarSet* [414]

⁵<https://www.pgmusic.com>

⁶<https://www.irealpro.com>

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

Resource	Link
ChoCo dataset	http://w3id.org/polifonia/resource/choco/
Portal page	https://smashub.github.io
JAMS Vocabulary namespace	http://w3id.org/polifonia/ontology/choco/ (prefix jams)
JAMS Resource namespace	http://w3id.org/polifonia/resource/choco/ (prefix pon-res)
Roman Chord Vocabulary namespace	http://w3id.org/polifonia/ontology/roman-chord/ (prefix roman)
GitHub organization & code	https://github.com/smashub/
Dataset generation code	https://github.com/smashub/choco
Documentation and tutorials	https://smashub.github.io/docs/category/choco-the-chord-corpus
Example data story	https://projects.dharc.unibo.it/melody/choco/chord_corpus_statistics
VoID description	https://github.com/smashub/choco/blob/main/void.ttl
SPARQL endpoint	https://polifonia.disi.unibo.it/choco/sparql
Zenodo	https://zenodo.org/badge/latest/doi/462698362

Table 4.2: Links to key ChoCo resources: ontology, datasets, and knowledge graph.

only provides 3 unique (and short) chord progressions. *UMA-Piano* [16] only contains audio recordings of chords, played independently. Finally, *POP909* [400] and the *Kostka-Payne* corpus [379] provide computationally-extracted chords and keys, whereas the first release of ChoCo focuses on high-quality annotations for time being.

From chordal data to JAMS datasets

The first challenge of bringing together existing chord datasets into a coherent, uniform corpus is the variety of formats in which chord annotations, and other related information, are encoded. In order to address this issue, we use JAMS data structure [203] as a simple, content-agnostic wrapper for expressing music annotations in general, and chord annotations in particular. JAMS relies on the popular Web data exchange JSON format, and enforces the following structure based on three basic properties⁷:

- **file_metadata**, describing the music piece these annotations refer to. More precisely, it contains these properties: **identifiers**, optionally providing explicit links to external resources, mostly relating to cataloguing information from online music databases, e.g., MusicBrainz⁸; **artist**, referring to a performer or a band; **title** of the musical work; **release**, intended as a more general definition of album; and **duration**, defining a temporal span within which annotations can fall.
- **annotations**, a container of annotation objects, each describing a specific

⁷<https://jams.readthedocs.io/>

⁸<https://musicbrainz.org>

namespace (the term *namespace* in JAMS has a different sense than a Web namespace) that identifies the type of the annotation’s subject (e.g., chords, structural segments, emotions, patterns, keys, etc.). These annotations also include metadata to document the annotation process (e.g. whether the annotation is manually produced or inferred by an algorithmic method, the name of the annotator or software, information about the annotation tools, rules and validation).

- **sandbox**, described as an unrestricted place to store any additional data.

Listings 4.1 and 4.2 show excerpts of an example JAMS file from the Isophonics collection [258] annotating chords for Queen’s *Bohemian Rhapsody*, taken from the Isophonics collection.

Although JAMS has an implicit focus for audio-based annotations, its definition and structure are flexible enough to be easily extendable to the symbolic domain. This is also confirmed by the modular design of the codebase, where additional namespaces can be registered by a user, by simply providing regular expressions to validate the annotation content (e.g. a new chord notation). In other words, any arbitrary music annotation can be described within JAMS as long as the atomic observations (e.g. the individual occurrences of chords making up the progression) are described in terms of: **time**, a temporal anchor specifying the onset of the observation; **duration**, **value** (e.g. *Bb:maj7*), and **confidence**, a scalar in $[0, 1]$ expressing a level of certainty by the annotator (or algorithm). Therefore, the only elements distinguishing audio from symbolic annotations, are the temporal specifications (time and duration), which are described in absolute (seconds) or metrical (measure and beat/offset) terms, respectively. For symbolic annotations, we number measures and beats from 1 for convenience, without attempting to emulate exact musical (editorial) practice for cases like anacrusic openings.

JAMification of datasets

Considering the diversity of annotation formats and conventions for data organisation (the way content is scattered across folders, files, database tables, etc.), each chord dataset in ChoCo (c.f. Table 4.1) undergoes a standardisation process leading to the creation of a JAMS dataset. This is needed to aggregate all relevant annotations of a piece (chord, key, etc.) in a single JAMS file, and to extract

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

content metadata from the relevant sources.

The content metadata of a (music) dataset is indeed crucial to identify, describe and retrieve the actual musical content being annotated. This typically includes the title of each piece, artists (composers and/or performers), and cataloguing information (album/release or collected work), ideally with the provision of identifiers (e.g. MusicBrainz IDs). Nevertheless, only the *Mozart Piano Sonata* collection [188] provides complete content metadata in a `csv` file, as usually expected from a music dataset. When content metadata is missing, this may be found online (HTML pages, supplementary material), from articles/reports documenting the collection, by resolving any cross-reference among files and dataset-specific identifiers, extracted from the actual score (or better, the dataset-specific representation of the score). Alternatively metadata can be derived from the organisation of files in folders. For example, `Michael Jackson/Essential Michael Jackson [Disc 01]/1-16_Beat_it.lab` indicates author, album, disc, track number and title, respectively. This organisation varies as the datasets vary – a consequence of the lack of a standard “datasheet for datasets” in the music domain [152].

```
1 {
2   "sandbox": {},
3   "annotations": [
4     {
5       "data": [
6         {
7           "duration": 0.459,
8           "confidence": 1.0,
9           "value": "N",
10          "time": 0.0
11        },
12        {
13          "duration": 3.663,
14          "confidence": 1.0,
15          "value": "Bb:maj6",
16          "time": 0.459
17        },
18        {
19          "duration": 0.789,
20          "confidence": 1.0,
21          "value": "C:7",
22          "time": 4.122
23        },
24        ...
25      ],
26    },
27  ]
28 }
```

Listing 4.1: *Excerpt of the three first chords in a JAMS file annotating Queen’s Bohemian Rhapsody.*

```
1   "annotation_metadata": {
2     "annotation_tools": "",
3     "curator": {
4       "name": "Matthias Mauch",
5       "email": "m.mauch@qmul.uk"
6     },
7     "annotator": {},
8     "version": 1.0,
9     "corpus": "Isophonics",
10    "annotation_rules": "",
11    "validation": "",
12    "data_source": ""
13  },
14  "namespace": "chord",
15  "sandbox": {}
16 }, ... ],
17 "file_metadata": {
18   "jams_version": "0.2.0",
19   "title": "Bohemian Rhapsody",
20   "identifiers": {},
21   "release": "",
22   "duration": 358.293,
23   "artist": "Queen"
24 }
25 }
```

Listing 4.2: *Annotation and file metadata in a JAMS file annotating Queen’s Bohemian Rhapsody.*

The same issue applies to the extraction, pre-processing, and standardisation of harmonic annotations from these collections, some of which were never released as chord datasets (*Weimar Jazz Database*, *Wikifonia*, *iReal Pro*, *Nottingham*). Harmonic annotations can be encoded in different formats (LAB, XLAB, RomanText, CSV, DCMLab, JSON, SQL, TXT), or extracted from symbolic music (MusicXML, ABC) and backing tracks in proprietary encodings (iReal, MGU). As each collection shows a specific combination of the mentioned issues (different organisation of content and metadata, different annotation formats and conventions), this step required considerable effort. The result of this standardisation process may improve the usability of these resources for music researchers, and simplifies the KG construction process. In addition, for the symbolic subsets, we also include time signatures (initial time signature and subsequent metrical changes) as annotations in each JAMS file (using a dedicated `timesig` namespace); which makes it easier to interpret the temporality of each chord (onset and duration) at hand.

Following the standardisation process, each of these 18 JAMS datasets represents a novel contribution per se, due to the heterogeneity of annotation formats and practices, and the limited availability of content metadata in their original version. This also includes *CASD*, a collection that provides chords in JAMS format, but lacks local key annotations, which were retrieved from *Billboard* (we remind that CASD is already a subset of Billboard).

Conversion of chord notations

As shown in Table 4.1, the third element of divergence besides annotation formats and provision of content metadata, is the notation system used to represent chords. To address this issue we perform the following actions: (i) decomposition of domain-specific notations to chord constituting elements; (ii) conversion of the decomposed chord to the Harte framework; (iii) conversion of chord progressions by iteratively applying steps (i) and (ii) to all the chords in a sequence/progression. This yields a new JAMS file with the converted chord annotations.

For all the above steps, specific software was developed for processing the different annotation types contained in the original datasets. There are three main types of chords that are processed: Roman Numerals chords (e.g. `C min:viio7/V`), Polychords (e.g. `E4,G#4,B4`), Leadsheet chords (e.g. `Gm7/F`). With *Leadsheet chords* we refer to a broader category, although each dataset using this format

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

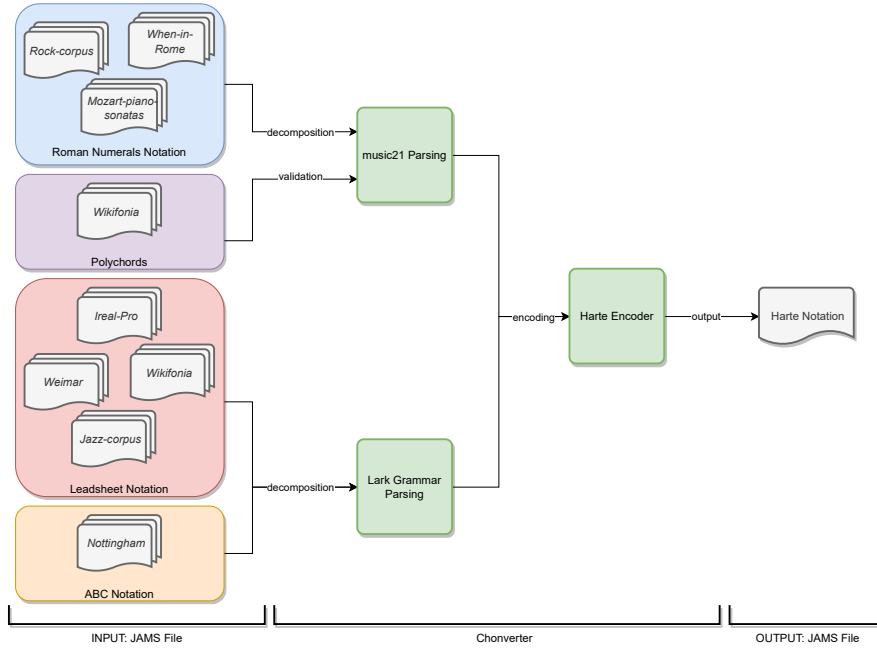


Figure 4.3: Overview of the Chonverter workflow, describing how different chord notations are converted to the Harte notation.

proposes a different flavour of this notation. For example, a G minor chord in *Wikifonia* is annotated as `G min`, whereas the same chord is annotated as `G-` in the *Jazz-corpus*.

As outlined in Table 4.1, each dataset uses a flavour of the same notation to represent chords, with the exception of *Wikifonia*, where some annotations use both Leadsheet and Polychords even for the same progression; and the Jazz Corpus, providing chords encoded in both Roman Numerals and Leadsheet. Figure 4.3 provides a taxonomy of the different notational flavours, together with a schematic overview of the conversion workflow.

In step (i), a chord is first decomposed into its components (e.g. C major \rightarrow C, E, G). For this purpose, the **Chonverter** uses a family of tools depending on the source notation. Roman numerals are decomposed using the *roman module* of **music21** [79], a Python library for computational musicology. As Polychords already provide note constituents by definition, this step is limited to preprocessing the symbols associated to the different pitches in a chord. Polychords are usually mixed with chords annotated in other notations (e.g. Leadsheet), so it is necessary to differentiate the type of chords when parsing. Finally, for each Leadsheet flavour, a context free grammar was created to parse the original annotation of the chord. A different grammar was created for each dataset containing annota-

tions in leadsheet format, namely *Weimar Jazz Database*, *Wikifonia*, and *iReal Pro*, using the `Lark` library⁹. Notably, the ABC notation used in *Nottingham* is similar to the Leadsheet notation and was therefore processed in the same way. This process is more intuitive for all collections natively using the Harte notation, as the latter already accounts for the description of chord pitches [181].

After all chords are decomposed as lists of pitches, it is then possible to associate a shorthand (a string) to each list according to the Harte notation (Step (ii)). The `Chonverter` achieves this via `music21` and defines rules for composing Harte chords.

New JAMS files are produced after the last step, each providing a new annotation (with `chord_harte` as namespace). Whenever an original annotation uses *Leadsheet* or *Polychord* notations, the new annotation replaces the original, since the conversion provides a generalisation of the different flavours via a syntactic transformation. Instead, if the original annotation contains *Roman Numerals* chords, the new (converted) annotation is added to the existing one, since the Roman Numerals contain information that would otherwise be lost, i.e. the harmonic functions that the chords hold within the piece.

The `Chonverter` module performs a syntactic conversion of chord labels. However, converting Roman Numeral also requires taking into account the key of the song. Moreover, a distinction has to be made between key-relative and absolute chords. Some music is always played in the same key, while other pieces are frequently transposed. For example, symphonies are often performed in a fixed key, while lieder are typically performed in multiple keys depending on the singer’s vocal range. Datasets like *When in Rome* contain transcriptions of these key-flexible works. Even in these cases, chords in ChoCo are always converted by taking into account the tonality provided by the original dataset for that piece. However, whenever this happens, the generated conversion, although correct, may only be one of several possible conversions.

The JAMS Ontology and the ChoCo Knowledge Graph

To represent JAMS annotations as Linked Data (LD) we designed an ontology that formally represents the JAMS data model. The JAMS Ontology is part of the Polifonia Ontology Network (c.f. Chapter 3), from which we reused 4 ontology modules (*Core*, *Music Meta*, *Music Representation* and *Music Projection*). More

⁹<https://github.com/lark-parser/lark>

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

specifically, the JAMS Ontology is part of the Music Annotation module, which directly imports it. Table 4.2 provides links to ChoCo’s resources, including the JAMS Ontology and KG.

The JAMS Ontology formally defines the semantics of music annotations that are encoded using JAMS. To improve compliance with the ontology and facilitate the generation of LD, we have established conventions for including relevant information in the creation phase of the JAMS files. In essence, the JAMS Ontology tackles the limitations of the current JAMS model, mainly on two fronts: (i) at the level of metadata, enabling the alignment and linking of tracks belonging to different datasets, and also, with external resources available on the Web; (ii) at the annotation level, allowing to describe data (e.g. a chord) by semantically annotating its components (e.g. root, quality, inversions, etc.) rather than using a label.

Concerning the first level, the JAMS Ontology inherits all the benefit of the proposed PON and Music Meta module, as documented in Section 3.5. Moreover, the proposed model also allows to correctly interpret the content of the annotation with great level of detail, for example, modelling temporal information both in real time (seconds) and in beats, depending on the type of annotation at hand.

To achieve this, additional data is dumped by the *JAMifier* in the *Sandbox* of each JAMS file, and new annotation types were created by contributing new *namespaces*. The JAMS Ontology provides a common conceptual, formal model to interpret JAMS annotations and is available online at the following URI:

<https://w3id.org/polifonia/ontology/jams/>

Our ontological requirements can be summarised as follows:

- the resulting KG must represent JAMS files and JAMS annotations as such, including their provenance and process-related information: e.g. source dataset, annotator, confidence of each observation, etc;
- temporal information must be expressed according to the type of the annotation’s subject, i.e. audio or score;
- chords must be represented according to the data model of these notation families: *Harte* and *Roman Numerals*.

To model this ontology, we reused the *Music Annotation Pattern* [94], an ODP [147] for modelling different types of music annotations and their related time

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

ID	Competency question
CQ1	What is the content of the observations contained in a JAMS Annotation?
CQ2	Who is the composer of a musical object?
CQ3	Who is the performer of a musical object?
CQ4	Who/what is the annotator of an annotation/observation, and what is its type?
CQ5	What is the time frame addressed by an annotation, within a musical object?
CQ6	What is its start time (i.e. the starting time of the time frame)?
CQ7	Which are the observations included in an annotation?
CQ8	Given an observation, what is the starting point of the time frame it addresses, within its target musical object?
CQ9	Given an observation, what is its addressed time frame, within its target musical object?
CQ10	What is the key of a composition/performance?
CQ11	What is the value of an observation?
CQ12	What is the confidence of an observation?
CQ13	What are the chords of a composition/performance?

Table 4.3: *Competency questions (CQs) addressed by the JAMS Ontology.*

references. We remark that the terminology used in the JAMS documentation¹⁰ is adopted to define the JAMS Ontology vocabulary. In particular, the following terms are (re-)used:

- **Annotation:** an annotation is defined as a group of **Observations** (see below) that share certain elements, such as the method used and the type of annotation’s subject (e.g. chords, notes, patterns);
- **Observation:** an observation is defined as the content of an annotation, and includes all the elements that characterise the observation. For example, in the case of an annotation containing chords, each observation corresponds to a chord, and specifies, in addition to the chord value, the temporal information and its confidence.

We also apply the same methodology described in Section 3.3: the CQs defined for the ontology are listed in Table 4.3, while the corresponding SPARQL queries are available in the JAMS Ontology repository¹¹.

Figure 4.4 shows a fragment of the JAMS Ontology modelling a JAMS Annotation. On the left (*box A*), we define the classes and properties for representing the song’s metadata, by reusing the *Music Meta* module from PON. Main classes and properties of this ontology are detailed in Section 3.5.

The connection between the Music Meta ontology and the JAMS Ontology happens at the level of `mm:Recording`, `mm:Score`, and `mm:AbstractScore`, where the union of the three form a `mr:MusicContent`, which can be annotated by a `jams:JAMSAnnotation`.

A core class of the JAMS Ontology is `jams:JAMSAnnotation`. It captures the

¹⁰<https://jams.readthedocs.io/en/stable/>

¹¹<https://github.com/polifonia-project/jams-ontology>

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

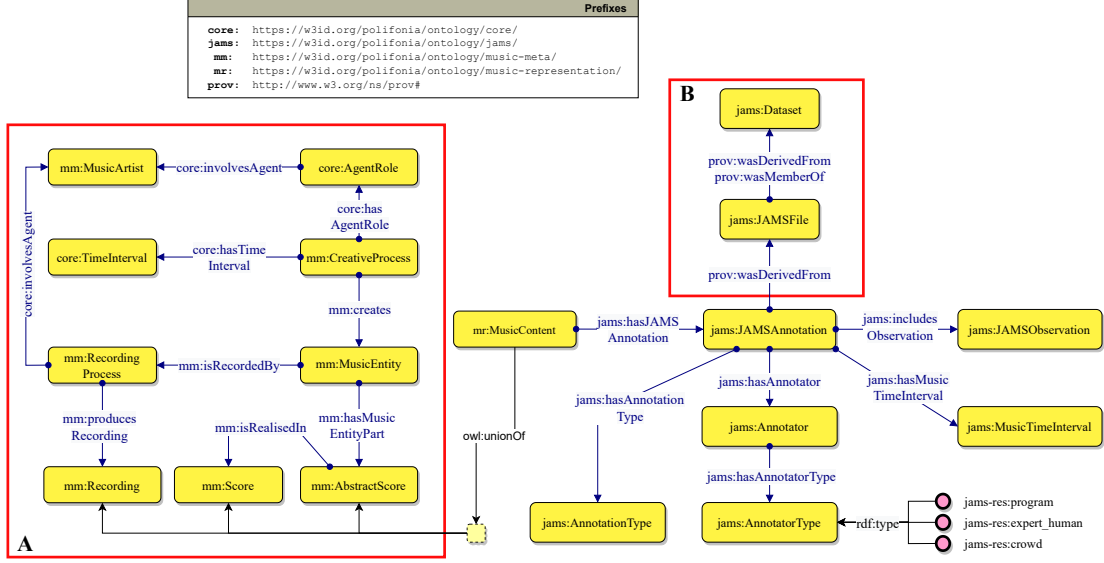


Figure 4.4: Fragment of the JAMS ontology describing JAMS files and their provenance, musical objects and JAMS annotations.

annotation, from a file encoded with the JAMS format, on a musical object (its target): either a recording or a score. A JAMS annotation entity and its musical object are put in relation by means of the property `jams:hasJAMSAnnotation`. An annotation is performed by an annotator `jams:Annotator`, has a time validity `jams:hasMusicTimeInterval`, and contains information of a certain type `jams:AnnotationType` (e.g. chords, keys, etc.). The validity indicates to which time frame, within a musical object, the annotation refers. For example, if an annotation reports the observation of a certain *key*, that *key* refers to a segment of the target musical object. Annotators may be of different types (e.g. expert annotator, software program), and are defined by the class `jams:AnnotatorType`. Finally, a `jams:JAMSAnnotation` is composed of a set of `jams:JAMSObservation`. Figure 4.5 depicts the JAMS Ontology fragment that models JAMS observations.

The Provenance Ontology [238] is reused to model the provenance of JAMS annotations (Figure 4.4, *box B*). Each JAMS annotation derives from a JAMS file (`jams:JAMSFile`) which is either taken or derived (for example, translated from a file in a different format to the JAMS format) from a dataset `jams:Dataset`.

A key aspect of observations and annotations is the identification of the musical object fragment they refer to. We model musical object fragments as musical time intervals `core:MusicTimeInterval`. Musical time intervals can be expressed in different ways, depending on the type of musical object. For example, if the subject of an observation (and in turn of an annotation) is a recording, then we

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

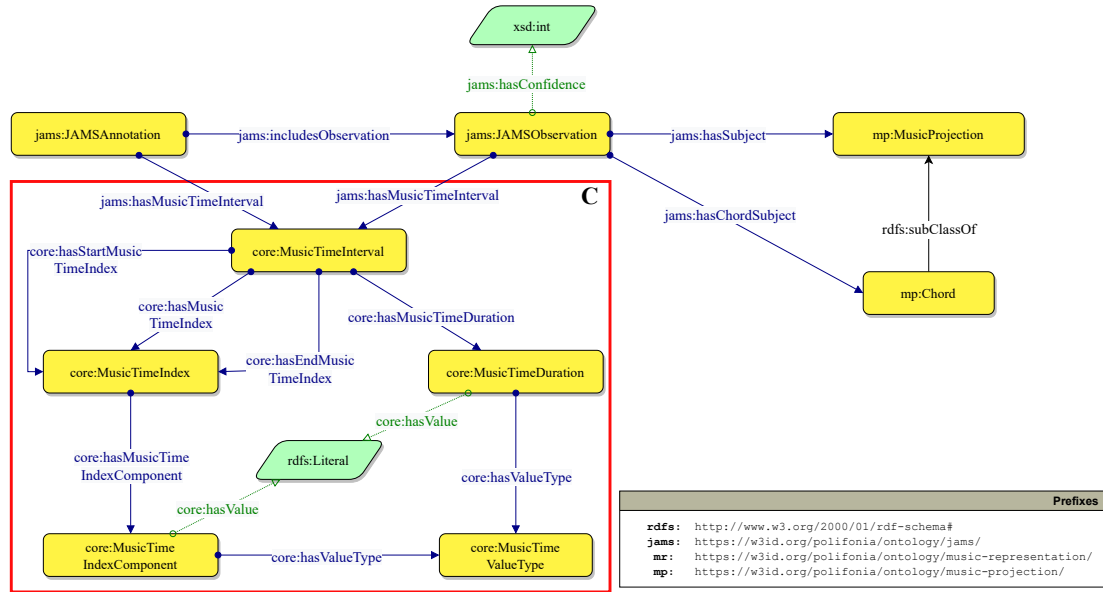


Figure 4.5: *Fragment of the JAMS Ontology describing JAMS annotations and JAMS observations. The red block C highlights how the time information has been modelled for handling different types of formats and standards.*

most probably identify its fragments in terms of seconds. If we deal with scores, we may want to use a combination of measures and beats. To make the ontology as flexible as possible for expressing musical time intervals, we model them as being defined by musical time indexes (`core:MusicTimeIndex`). Each musical time interval has a start time index and an end time index (plus potentially infinite internal time indexes). A musical time index is defined by one or more components (`core:MusicTimeIndexComponent`), each substantiated by a value (`core:hasValue`) and a value type (`core:MusicTimeValueType`). A musical time interval also has a duration (`core:MusicTimeDuration`) which is expressed by means of a value and a value type (usually seconds for recordings and beats for scores).

Figure 4.6 shows an example of data from the *Wikifonia* subset (*wikifonia_39*) annotated using the *JAMS Ontology*. Starting from the individual highlighted by the red box (`pon-res:AutumnInRomeComposition`) we can trace information related to the piece entitled "*Autumn in Rome*". The file includes two annotations (`Wikifonia39KeyAnnotation` and `Wikifonia39ChordAnnotation`), derived from a score, hence their temporal information is expressed as a combination of **beats** and **measures**. The chord annotation (`pon-res:Wikifonia39ChordAnnotation`) contains two observations, the first starting at the beginning of the first measure

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

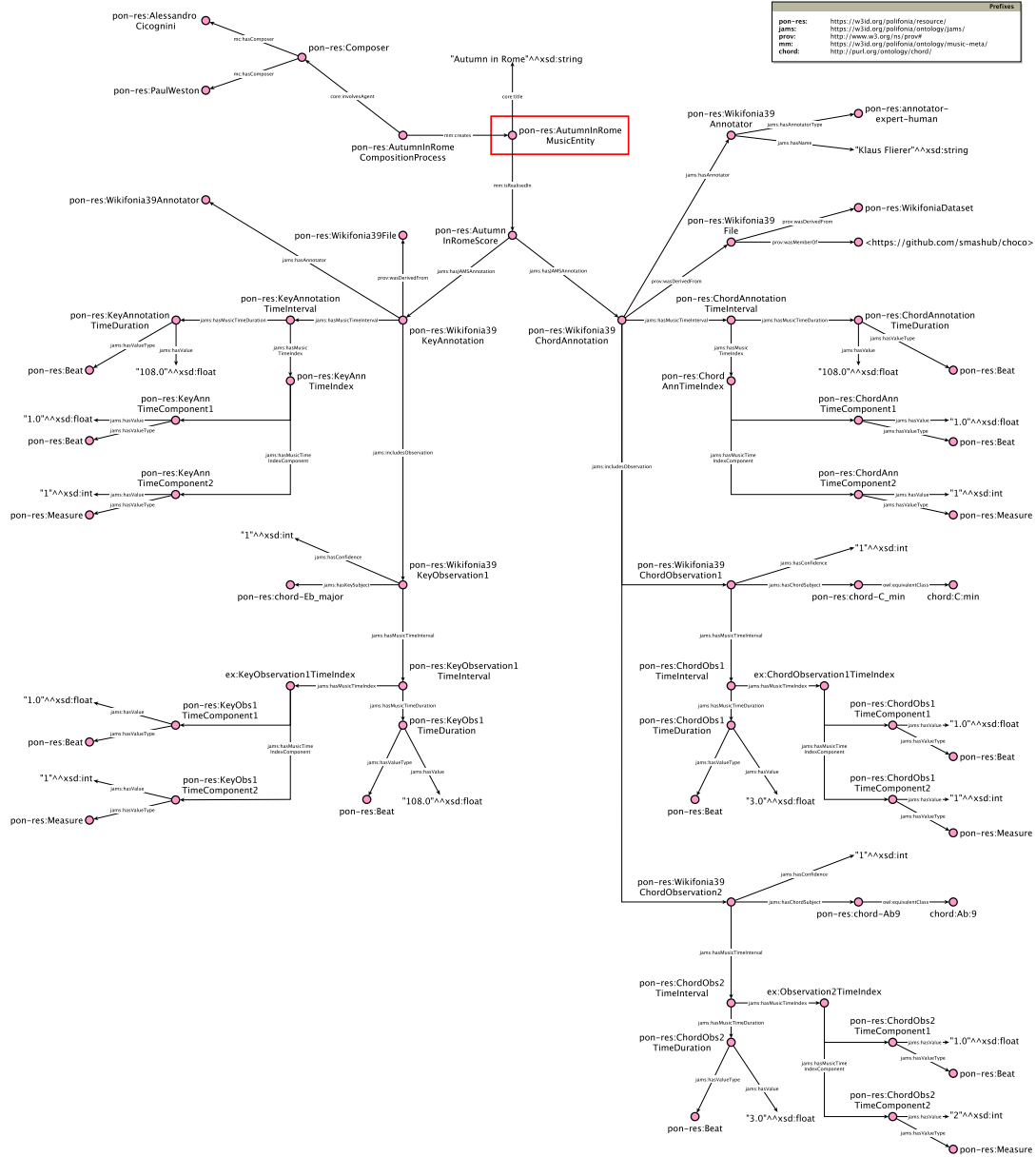


Figure 4.6: Example of data modelled using JAMS Ontology, extracted from a track from the Wikifonia dataset. The track is annotated from a score, therefore annotations and observations contain time references expressed in beat and measure.

(*measure 1, beat 1*), while the second starts at the beginning of the second measure (*measure 2, beat 1*). They both have a duration of 3 *beats*.

We remark that the music time interval of an annotation is different, though dependent on, the time interval of its observations: it must include all of them. In the example of Figure 4.6, the time interval of `pon-res:Wikifonia39ChordAnnotation` starts from the beginning of the first measure (*measure 1, beat 1*) and has a duration of 108 *beats*.

A JAMS observation, according to the JAMS data model, can only have one subject (`jams:hasSubject`), which is a music projection (`mp:MusicProjection`) e.g. chord, key mode, pitch. The main musical feature currently treated in ChoCo is the chord. A chord (`mp:Chord`) is indeed modelled as a special type of `mp:MusicProjection`.

As presented in Section 4.3.1, ChoCo focuses on two chord notations: Harte and Roman Numerals. In the JAMS Ontology, the Harte notation is addressed by reusing and adapting the Music Projection module of PON, which classes were further aligned to the Chord ontology [374]. For modelling Roman Numerals, we developed the *Roman Chord Ontology*¹², which is part of the *Music Analysis* module in PON. Figure 4.7 shows the main features of the ontology, which is available at the following URI:

<https://w3id.org/polifonia/ontology/roman-chord/>

The core class `roman:Chord` defines roman numeral chords. A chord is a complex structure, therefore it is described by means of several properties. The classes `roman:BasicFunction` and `roman:Quality` describe the chord from a functional harmony perspective and the quality of the chord (e.g. major, minor, augmented), respectively. The class `roman>Note` describes the absolute pitch of the bass note, while the class `roman:Interval` is used to describe the bass, the internal intervals of the chord and any missing intervals. Each interval is described by the datatype properties `roman:hasDegree`, which describes the degree of the interval, and `roman:hasModifier`, which describes any alterations to the interval. Finally, the datatype property `roman:inversionType` defines the possible type of inversion of the chord.

To streamline this process and simplify its reuse, we also release service APIs allowing to generate knowledge graphs of roman numeral chords – starting from their symbol and a reference key. The API service can be queried as follows:

¹²<https://github.com/polifonia-project/roman-chord-ontology>

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

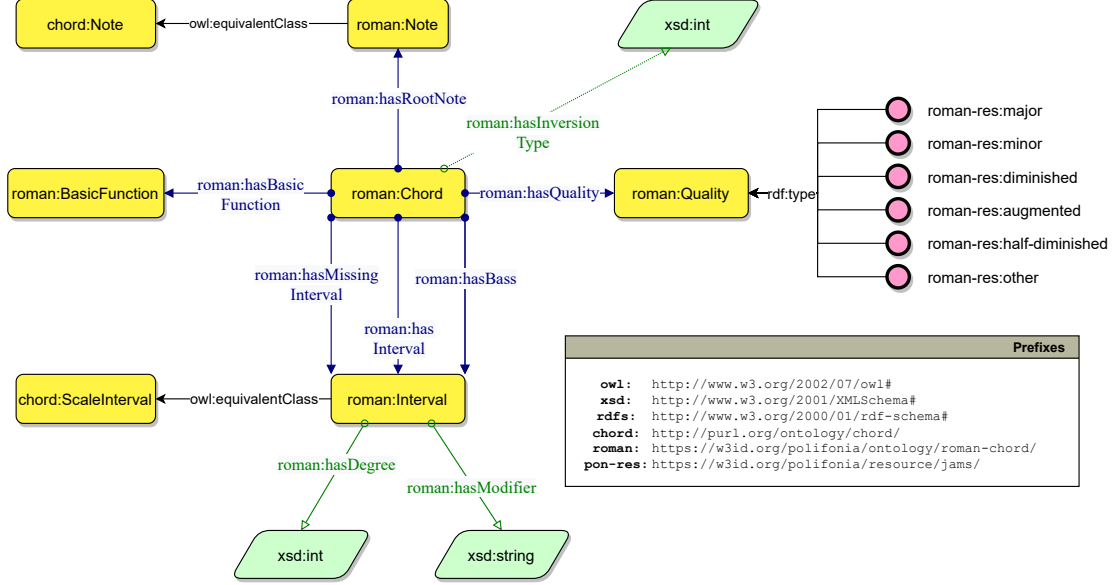


Figure 4.7: The Roman Chord Ontology describing Roman Numeral Chords and their constituting elements.

[https://w3id.org/polifonia/resource/roman-chord/\[romanChord\]_\[key\]](https://w3id.org/polifonia/resource/roman-chord/[romanChord]_[key])

For example, an API call where the IV53[no3]_C is requested, will return the knowledge graph illustrated in Figure 4.8.

Knowledge Graph construction. To build the ChoCo Knowledge Graph (ChoCo KG) we propose `jams2rdf`, an open-source tool to convert any JAMS file to RDF, with the following usage:

```
jams2rdf.py <input_jams_file> [<outout_rdf_file>].
```

`jams2rdf` relies on *SPARQL Anything* [81], a tool supporting querying with SPARQL any data from any file format. We use SPARQL Anything’s JSON module to define a SPARQL CONSTRUCT query template that generates ChoCo triples according to the JAMS Ontology (Figure 4.4). This allows for a modular design, as different conceptualisations, ontologies and triplifications for JAMS can be added in separate, independent SPARQL queries. We also publish additional queries to facilitate the extract and the manipulation of specific JAMS fields from the KG.

To build the ChoCo KG, we iteratively run `jams2rdf` using the query template over our entire collection of curated JAMS files. This yielded ≈ 30 milion RDF triples.

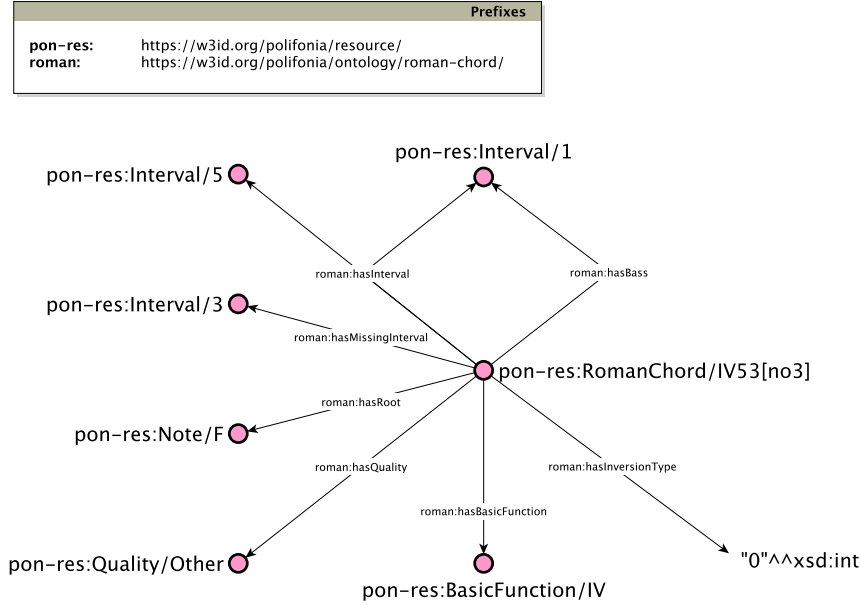


Figure 4.8: Example of a Knowledge Graph generated using the Roman Chord Ontology on a *IV53[no3]* chord.

More statistics on the ChoCo KG can be found in the Melody portal of the Polifonia Project¹³.

4.3.2 Data Records

The descriptive statistics reported in this section provide an overview of ChoCo at two different levels: the metadata associated to the music tracks and scores in the dataset (the musical content being annotated), including their identifiers and links; and the actual content of the music annotations.

In ChoCo v1.0 [90] (from now on, ChoCo), the dataset contains 20,086 JAMS files: 2,283 from the audio collections, and 17,803 collected from symbolic music. In turn, these JAMS files provide 60,263 different annotations: 20,530 chord annotations in the Harte notation (c.f. Section 4.3.1), and 20,029 annotations of tonality and modulations – hence spanning both local and global keys, when available. Besides the harmonic content, ChoCo also provides 554 structural annotations (structural segmentations related to music form) and 286 beat annotations (temporal onsets of beats) for the audio subsets.

¹³https://projects.dharc.unibo.it/melody/choco/chord_corpus_statistics

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

Metadata and external links

The average duration of the annotated music pieces is 191.29 ± 85.04 seconds for (audio) tracks; with a median of 104 measures for symbolic music, and Interquartile range $IQR = Q3 - Q1 = 168 - 42 = 126$ (Q1, Q3 denote first and third quartiles, respectively). As illustrated in Figure 4.9, this provides a heterogeneous corpus with a large extent of variability in the duration of pieces, which also confirms the diversity of musical genres in ChoCo (Table 4.1). For instance, a folk tune can span a few measures and still possess a musical identity with respect to the genre; in contrast, a sonata can cover hundreds of measures.

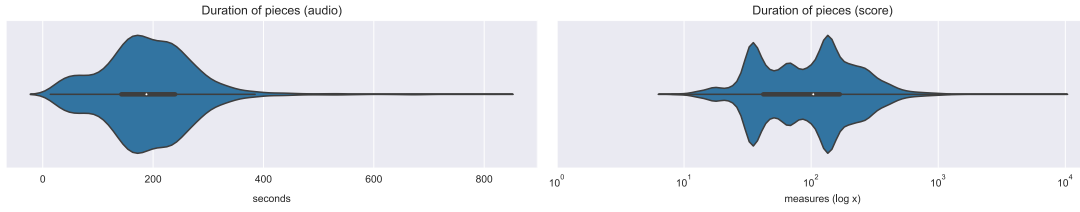


Figure 4.9: Distribution of audio track (left) and score (right, log-x scale) durations, measured in seconds and measures, respectively.

From the metadata extraction of the JAMification step (c.f. Section 4.3.1), it was possible to disambiguate 2421 artists as *performers* – which represent 12.05% of the dataset, and a total of 7,304 as *composers* (36.36% of ChoCo). This implies that the remaining 51.59% of JAMS files only provide generic *artist* information (with no distinction between composers and performers), whereas another small portion of the dataset – corresponding to the JazzCorpus (0.37% of ChoCo), does not provide any metadata. An overview of the ten most common performers and composers is reported in Figure 4.10, with “*The Beatles*” and “*Franz Schubert*” being the most recurring names, respectively.

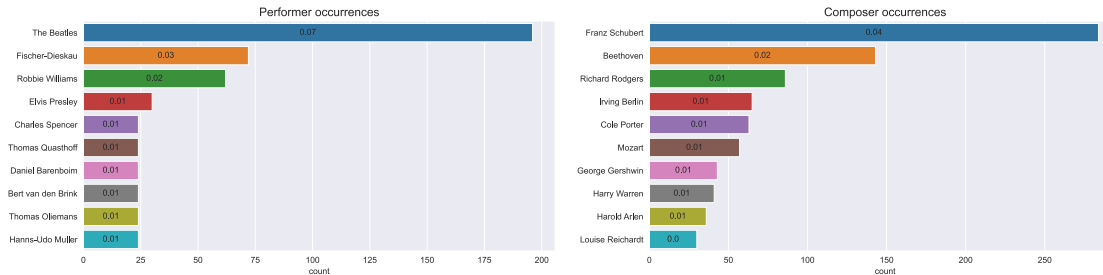


Figure 4.10: Overview of the ten most common performers (left) and composers (right) in the dataset, when explicitly distinguishable from their generic “artist” attribution in the metadata.

The JAMS files in ChoCo also contain 771 links to other resources, representing about 3.8% of the dataset. These were extracted from the original collections, and automatically verified and corrected for validity (link/identifier working) and consistency (disambiguation of the resource pointed, e.g., musical work, recording, and release). Most links point to MusicBrainz (78%), whereas a few of them link to Wikidata (6%), IMSLP (6%), YouTube (5%), and to other datasets (5%).

In addition to these explicit links, which can already be found in the JAMS files, we also link the resources in the ChoCo KG to two other large-scale music datasets on the Web:

1. **MIDI Linked Data Cloud** [272]. The ChoCo chord annotations can be useful for harmonic analyses of existing scores and symbolic music representations, e.g. MIDI. To link MIDI URIs with ChoCo URIs, we compare the string similarity of the original MIDI filename and the JAMS `file_metadata` name, both typically containing the band/artist and song names, and link them through `midi:midiOf` if their similarity is $>.80$. This yields 2,411 links. However, we do not inspect the musical content to establish this linkage, meaning that the harmonic annotation of a sonata in C minor would be linked to the same sonata in D minor, as long as their titles are highly similar. Therefore, the verification and the provision of links that are musically plausible (beyond the metadata) are currently under investigation.
2. **Listening Experience Database (LED)** [4]. Relating harmonic properties of pieces and their evolution to music listening experiences throughout history is also another promising direction. For those listening experiences that are explicitly associated to a musical work through `dc:subject` and `mo:performance_of` (where `dc` and `mo` prefix *Dublin Core* and *Music Ontology*, respectively), we extract links with ChoCo's resources via text similarity of work titles (using the same criteria as before). Links can be further filtered whenever a musical work in LED also provides a reference to the artist (via `mo:composer` or `mo:performer`). Overall, this yields 1996 links.

These additional links open up new research directions, as they allow to relate harmonic content (chord changes, harmonic complexity, tension, etc.) to other musical properties that are inherently present in the music (melodic contour, expressive variations, instrumental changes, etc.), or that may have been elicited certain emotions, memories, and feelings in listeners. Here we report an example

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

of a listening experience of “*So What*” in LED¹⁴, which was linked to 8 chord annotations in ChoCo.

«What do you mean by playing "without harmony"? Using a pedal tone, which Coltrane got into after a period of very dense harmonic playing. He would use one or two harmonic references throughout a song, as he did on “So What” [from Miles Davis’s Kind of Blue, on Columbia]. It was basically D for sixteen bars, E flat for eight bars, and then back to D. Ultimately, he worked with only one harmonic reference point, and then in “Ascension” [from Best of John Coltrane: His Greatest Years, on Impulse] there was nothing harmonically.» (Steve Kuhn in “The Great Jazz Pianists: Speaking of Their Lives and Music”)

Overview of chordal annotations

This section provides statistics on the content of chord annotations in ChoCo, their observations and temporal onsets; similar statistics can also be extracted for tonality annotations (local and global keys), but are excluded here to focus on chordal content.

Overall, and without any simplification/collapsing of chords, there are 1,575,409 chord occurrences/observations in ChoCo, with an average annotation having 76 chords (Figure 4.11, *left*). When looking at the unique chord occurrences in the harmonic progressions (chord classes) – measuring the chordal diversity of the annotations, the dataset counts 306,407 chords, which are drawn from a set of 7,281 possible classes. An annotation, on average, uses 14.92 ± 11.10 chord classes (Figure 4.11, *right*). The median duration of chord observations in audio and score JAMS is 1.6 (Q1 = 1.12, Q3 = 2.15) seconds and 3.06 (Q1 = 2.33, Q3 = 4) beats, respectively (Figure 4.12). For most statistics reported in this section, we observe right skewed distributions (long tails on the right side) as negative values (e.g. negative durations) cannot occur; and we report log-x plots for convenience.

The fifteen most common chords in ChoCo, based on their absolute and relative occurrences, are reported in Figure 4.13 (*left*). Absolute counts are obtained by accumulating the chord counts for each annotation/progression across the dataset (as if all annotations refer to the same piece). Instead, relative counts are computed by first normalising the absolute counts of each annotation by the number

¹⁴<https://data.open.ac.uk/page/led/lexp/1431335026178>

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

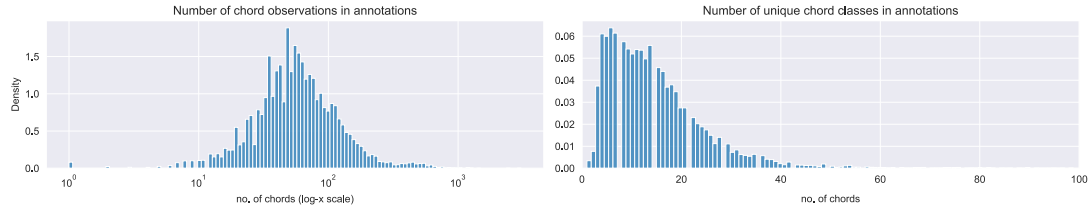


Figure 4.11: *Distribution of the number of chord observations per annotation (left, linear scale) and their distinct chord classes (right, log-x scale). The latter can also be considered as the cardinality of the chord set used by each annotation.*

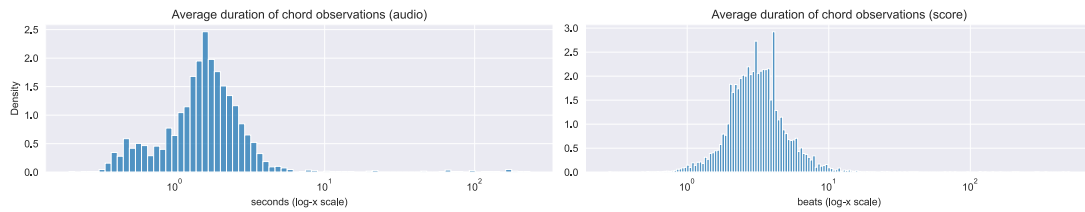


Figure 4.12: *Distribution of chord durations for audio (left, seconds) and symbolic (right, beats) annotations on log-x scale.*

of chord observations in the progression; then averaging the resulting chord frequencies across all annotations. Analogously, Figure 4.13 (*right*) reports the same statistics after removal of consecutively repeated chords. This pre-processing step aims to mitigate consecutive repetitions (which may arise due to the different temporal granularity of chord observations, or possess a harmonic function) from inflating the chord count. Regardless of the counting method, the three most common chords in the dataset are: *C:maj*, *G:maj*, and *F:maj*.

A similar analysis is also reported for chord n-grams, which are typically used

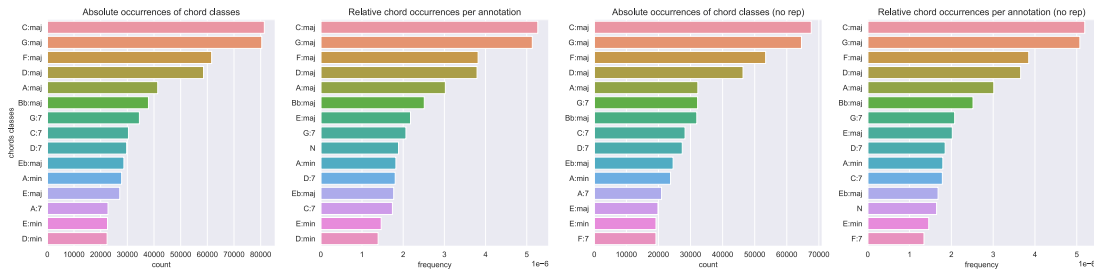


Figure 4.13: *Absolute and relative occurrences of chord classes in the original annotations (left, centre-left), and after removal of consecutively repeated chords (right, centre-right). Absolute occurrences are counted and accumulated throughout the corpus, whereas relative occurrences are first aggregated per annotation, as frequencies, then averaged across the whole dataset. Note that the “N” chord class denotes the “silent chord” as per the Harte notation (obtained for all subsets).*

4.3. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs

to find harmonic patterns in songs. To avoid trivial n-grams, these are computed after removal of consecutive repetitions (e.g. G:7, G:7, C:maj becoming G:7, C:maj). Table 4.4 ranks the first 10 n-grams based on their relative count (frequency).

To conclude, the number of chord annotations for which the identity of the annotators is known is 796 (3.9% of the dataset).

Order	Rank	Chord 1	Chord 2	Chord 3	Chord 4	Frequency	Occurrences
2	1	G:maj	C:maj	-	-	9.894371e-07	11560
	2	C:maj	G:maj	-	-	9.314316e-07	9968
	3	C:maj	F:maj	-	-	8.578674e-07	9837
	4	D:maj	G:maj	-	-	8.447899e-07	11229
	5	G:7	C:maj	-	-	8.270923e-07	12590
	6	G:maj	D:maj	-	-	8.236944e-07	9591
	7	F:maj	C:maj	-	-	7.588854e-07	8547
	8	D:7	G:maj	-	-	7.092709e-07	10673
	9	A:maj	D:maj	-	-	6.319998e-07	6925
	10	C:7	F:maj	-	-	6.247398e-07	10362
3	1	G:maj	C:maj	G:maj	-	4.156081e-07	4487
	2	C:maj	F:maj	C:maj	-	4.022300e-07	4167
	3	D:maj	G:maj	D:maj	-	3.518498e-07	4473
	4	C:maj	G:maj	C:maj	-	3.210295e-07	3209
	5	G:maj	D:7	G:maj	-	2.757892e-07	3411
	6	G:maj	D:maj	G:maj	-	2.755515e-07	3483
	7	C:maj	G:7	C:maj	-	2.685492e-07	3371
	8	F:maj	C:maj	F:maj	-	2.601499e-07	2660
	9	A:maj	E:maj	A:maj	-	2.201239e-07	1767
	10	A:maj	D:maj	A:maj	-	2.151695e-07	2450
4	1	G:maj	C:maj	G:maj	C:maj	1.984606e-07	1933
	2	C:maj	G:maj	C:maj	G:maj	1.897574e-07	1746
	3	C:maj	F:maj	C:maj	F:maj	1.840459e-07	1693
	4	F:maj	C:maj	F:maj	C:maj	1.759950e-07	1509
	5	D:maj	G:maj	D:maj	G:maj	1.647309e-07	2256
	6	G:maj	D:maj	G:maj	D:maj	1.609514e-07	2105
	7	D:7	G:maj	D:7	G:maj	1.587393e-07	1873
	8	A:maj	E:maj	A:maj	E:maj	1.497483e-07	998
	9	E:maj	A:maj	E:maj	A:maj	1.453102e-07	1067
	10	G:7	C:maj	G:7	C:maj	1.338413e-07	1593

Table 4.4: Summary of the most common chord n-grams ($n = 2, 3, 4$), ranked by their relative occurrence (frequency) per chord annotation. The last column reports the corresponding total number of n-gram occurrences in the dataset (no aggregation).

4.4 Technical Validation

To validate the data transformation workflow presented in Section 4.3.1 (Figure 4.1), focusing on the output of the **JAMifier** (generation of JAMS files from arbitrary chord collections) and the **Chonverter** (chord alignment and conversion) modules, we conducted two separate analyses: a groundtruth evaluation of JAMS files, and an expert validation of chord conversions.

4.4.1 Validation of the JAMifier

As the goal of the JAMifier is to automatically generate a JAMS dataset given a music collection providing chord annotations and metadata in different formats, notations, and conventions, this first evaluation addresses the following question.

How complete and accurate are ChoCo’s JAMS files – for metadata and harmonic annotations, after the JAMification?

To answer this question, we carried out a series of tests to compare a sample of generated JAMS files with those that are expected from this process. This required the creation of a groundtruth dataset of JAMS files that were manually produced by two human annotators from a given template (the backbone of a JAMS file), and through manual inspection of the original collections. For example, given a sample of the Wikifonia subset, the validator was expected to fill the JAMS template by: opening the MusicXML file of each assigned piece; inserting the relevant metadata (title, composer, duration, etc.) into the appropriate fields; and finding the (Leadsheet) chord labels annotated on the score – to create a JAMS **Observation** out of each of them. Annotators were first instructed on the task, and a preliminary annotation trial was performed to assess their reliability. After the trial, annotators received 4 templates for each subset and produced 72 gold JAMS files in total. The corresponding JAMification output is then compared to the groundtruth to measure: (i) the coverage and the accuracy of the metadata; and (ii) the coverage and error of chord and key annotations.

For the metadata, coverage is computed as the proportion of metadata fields in the gold JAMS that can also be found in the generated JAMS, regardless of their values. For example, if *title*, *composers*, *genre*, and *duration* are the expected metadata fields for a given JAMS file, and the generated counterpart only provides records for *title* and *duration*, coverage would account for 0.5 (even if both *title* and *duration* are incorrect). To provide a complementary view, metadata accuracy

4.4. Technical Validation

of common fields is computed as the normalised Levenshtein similarity among the generated and expected values for strings; or as the relative variance from the expected value for numerical fields (e.g. duration). The accuracies are then averaged for each JAMS file.

The results of this evaluation are reported in Table 4.5, aggregated for each subset and separated from the identifiers that were extracted from the JAMification (e.g. MusicBrainz, Wikidata – c.f. Section 4.3.2). Overall, maximum accuracy and coverage are attained for most collections, and all the possible identifiers are always extracted with no errors.

subset	metadatametadata		identifiersidentifiers	
	coverage \uparrow	accuracy \uparrow	coverage \uparrow	accuracy \uparrow
biab-internet-corpus	0.95 ± 0.1	0.9243 ± 0.0835	-	-
billboard	1.0 ± 0.0	1.0 ± 0.0	-	-
chordify	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
ireal-pro	1.0 ± 0.0	1.0 ± 0.0	-	-
isophonics	1.0 ± 0.0	1.0 ± 0.0	-	-
jaah	0.8036 ± 0.0595	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
jazz-corpus	1.0 ± 0.0	1.0 ± 0.0	-	-
mozart-piano-sonatas	0.875 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
nottingham	1.0 ± 0.0	1.0 ± 0.0	-	-
real-book	1.0 ± 0.0	1.0 ± 0.0	-	-
robbie-williams	1.0 ± 0.0	1.0 ± 0.0	-	-
rock-corpus	1.0 ± 0.0	1.0 ± 0.0	-	-
rwc-pop	1.0 ± 0.0	0.9999 ± 0.0001	-	-
schubert-winterreise	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
uspop2002	1.0 ± 0.0	0.9661 ± 0.062	-	-
weimar	1.0 ± 0.0	0.9878 ± 0.0243	1.0 ± 0.0	1.0 ± 0.0
when-in-rome	0.7976 ± 0.0558	0.9608 ± 0.0694	-	-
wikifonia	0.95 ± 0.1	0.95 ± 0.1	-	-

Table 4.5: Average coverage and accuracy of metadata and identifiers in the generated JAMS files, per ChoCo subset. The dash symbol denotes a subset that does not provide any identifiers.

For the harmonic annotations in the JAMS files, comparison with the gold counterparts is focused on *coverage* and *error* – reported independently for times (e.g. the onset of a chord occurrence), durations (e.g. how long a chord occurrence spans), and labels (e.g. a *C:maj* chord) of the observations in each annotation. The evaluation is thus in line with the structure of an observation in JAMS’ annotations (see Section 4.3.1 and Listings 4.1, 4.2). In this case, *coverage* measures the amount of the overlap between the generated and the expected observation

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

fields, without taking order into account (this is because an extra observation may have been inserted by the annotator, thus breaking the desired alignment for comparison). For example, if (C:maj, G:maj, D:7, F:maj) and (N, C:maj, G:maj, D:7) are the labels of a generated chord annotation and the corresponding gold, respectively, the silent chord “N” breaks the alignment of those sequences. In this case, coverage would still be $3/4$, as all the other chord labels are included in generated annotation. Instead, errors are computed from a 1-to-1 comparison of fields – which are assumed to be aligned. The latter can be reported according to the unit of measure of each field: seconds and beats for *time* and *duration*, and normalised Levenshtein distance for *labels* (string values).

subset	type	Key coverages ↑			Key errors ↓			Chord coverages ↑			Chord errors ↓		
		time	duration	label	time	duration	label	time	duration	label	time	duration	label
billboard	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
chordify	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
isophonics	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
jaah	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.95 ± 0.1	0.95 ± 0.1	1.0 ± 0.0	0.06 ± 0.13	0.06 ± 0.13	0.0 ± 0.0
robbie-williams	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
rwc-pop	audio	-	-	-	-	-	-	1.0 ± 0.0	0.53 ± 0.45	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
schubert-winterreise	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
uspop2002	audio	-	-	-	-	-	-	1.0 ± 0.0	0.3 ± 0.26	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
weimar	audio	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
biab-internet-corpus	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.95 ± 0.1	1.0 ± 0.0	1.0 ± 0.0	0.05 ± 0.1	0.0 ± 0.0	0.0 ± 0.0
ireal-pro	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
jazz-corpus	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
mozart-piano-sonatas	score	0.5 ± 0.58	0.0 ± 0.0	0.5 ± 0.58	62.55 ± 125.03	139.75 ± 83.75	0.25 ± 0.29	0.85 ± 0.3	0.88 ± 0.25	0.75 ± 0.5	0.25 ± 0.5	0.15 ± 0.19	0.15 ± 0.3
nottingham	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.75 ± 0.25	1.0 ± 0.0	1.0 ± 0.0	0.85 ± 0.6	0.0 ± 0.0	0.0 ± 0.0
real-book	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
rock-corpus	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
schubert-winterreise	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
when-in-rome	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
wikifonia	score	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.0 ± 0.0	0.92 ± 0.17	0.75 ± 0.35	0.0 ± 0.0	0.1 ± 0.2	0.11 ± 0.18

Table 4.6: Evaluation of chord and key annotations in the generated JAMS files on the test samples, reported for times, durations, and labels of their observations, and averaged for each subset. Coverage of observation values ranges from 0 (all the expected values are not found in the generated annotation) to 1 (all the expected values are included). Errors are given as seconds (audio) or beats (symbolic) for times and durations, respectively; and as normalised text similarities for labels.

Table 4.6 reports the results of this last evaluation for both key and chord annotations, where each metric is averaged by subset (mean and standard deviation). Results show good coverage and minimum error for most subsets, thus confirming the quality of the JAMification output. An exception is the Mozart Piano Sonata collection, for which low coverage and high errors are reported for key annotations. After having manually compared the JAMS sample for this subset, we found that the observations annotated by our validators in the gold set used a different temporal granularity (e.g. merging two consecutively repeated observations and aggregating their time and duration), compared to the JAMification output. Although this affected the evaluation results, both these annotations can be deemed equivalent.

4.4.2 Validation of the Chonverter

Following the data transformation workflow illustrated in Figure 4.1, we recall that the output of the JAMification step that does not natively provide Harte chord labels undergoes an alignment/conversion process through the **Chonverter**. First, the *Chonverter* aligns chord labels to one of the three chord families introduced in Section 4.3.1, namely: *Leadsheet* (Harte), *Roman*, and *Polychord*. Then, a syntactic conversion is performed on each chord class, independently, to infer the corresponding Harte label. Evaluating the output of the Chonverter can thus be formulated as follows.

How accurate and musically plausible are the chord alignment and chord conversion steps?

Conversely to the previous evaluation, addressing this question requires musical expertise and familiarity with different chord notations. Therefore, we performed a 2-step evaluation with music experts to validate the alignment and the conversion rules. Four participants with at least 5 years of musical training were recruited for this experiment. Participants were first introduced to the task, and asked to express their level of familiarity with the different chord notations, and the validation methodology. Given the nature of the validation, no personal record was recorded from participants and minimal risk clearance was granted from the Research Ethics Office of King’s College London (registration number: MRSP-21/22-32842).

Step 1 The first step focused on validating the context-free grammars used to parse chords in the original formats and aligning them to the corresponding chord families. Participants were presented with 3 different grammars, including 250 mapping rules to validate. Whenever a rule was deemed incorrect, participants were asked to provide the expected mapping.

Step 2 Once chords were converted, the final result of the conversion was validated. This step also allowed for the validation of other conversion types that were not validated in Step 1, such as Roman numerals and Polychords. In addition, even for annotations originally provided in Leadsheets, this step allows for the validation of added/removed notes and inversions.

The first step allowed to validate all the grammar rules used for decomposing leadsheet chords into their constituting degrees. Each grammar consists of a set

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

Subset	Validated chords	Chord type	Correct conversions	Incorrect conversions	Accuracy \uparrow
ireal-pro	39	leadsheet	37	2	0.949
rock-corpus	40	roman	40	0	1.000
weimar	37	leadsheet	37	0	1.000
when-in-rome	40	roman	40	0	1.000
wikifonia	40	leadsheet	39	1	0.975
<i>average</i>	<i>196</i>	<i>all</i>	<i>193</i>	<i>3</i>	<i>0.985</i>

Table 4.7: *Evaluation of chord conversions performed by music experts on a selection of ChoCo subsets.*

of *shorthands* grouped into classes. For example, the class referring to minor chords is composed of the shorthands "m" and "min". Each class is then mapped to the degrees that compose that type of chord: for minor chords, the degrees associated with that class are 1, b3, 5. This type of validation was required due to the limited musical background of the dataset’s curators. All grammar rules reported incorrect by the experts were corrected and revised. A total of 27 rules within the validated grammars were updated. The corrections were of two main types: i) *correct shorthands but incorrect degrees*: the group of shorthands assigned to degrees was correct, but the degrees into which the chord was decomposed had one or more errors; ii) *inconsistent group of shorthands*: the grouping of shorthands in classes was incorrect. In this case, the shorthand(s) not belonging to the class was moved to the correct class if it existed, otherwise a new class was created. This implies that the preliminary chord alignment of the **Chonverter** is potentially error free.

The second validation step consisted in distributing spreadsheets in which the original chords were shown in the first column whereas the second column showed the chords converted by the *Chonverter* module. Before starting this validation phase, all participants were provided with a thorough documentation of all types of annotation used, including Harte. Furthermore, chords annotated in the *Roman Numeral* format, which had not been validated in the previous step, were tested for the first time. Experts were asked to mark whether the conversion to the Harte format was correct or not. The evaluation results are as the percentage of corrected chords out of the total (Table 4.7).

4.5 Usage Notes

The availability of a large chord dataset, providing high-quality harmonic annotations with temporal information, content metadata, and links to external

resources, is of considerable interest to several research communities. In the field of MIR, chord datasets are a fundamental prerequisite for training and evaluating content-based music algorithms that can accommodate a variety of tasks – from chord recognition and cover song detection, to automatic composition systems. For musicology and computational music analysis, the scale and diversity of ChoCo [90] would enable large scale cross-corpus studies across different musical periods, genres, and artists (e.g. uncovering potential influences), and the KG can also be leveraged to run complex queries entailing certain musicological properties of chords, rather than relying exclusively on their notation-specific label. Also the SW community would benefit from the introduction of high quality chord data that can be linked to existing Web resources. In turn, this opens up new scenarios and research opportunities for the aforementioned communities.

4.5.1 Applications and tasks

Given the diversity, size, and quality of the corpus, we expect ChoCo to enable novel applications in Music Technology, other than supporting the design and the evaluation of methods addressing specific tasks in both MIR and computational music analysis. Besides the aforementioned applications in music listening and recommendation, another case study involves the advancement of systems for machine creativity. In the context of our work, these include automatic (or semi-automatic) composition, with particular focus on *arrangement generation* [358] (generating a chord progression, possibly given a melody to accompany); and *melody generation* through harmonic conditioning [128] (generating a melody to play along with a chord progression that is provided as a harmonic template). In ChoCo v1.0, this is enabled by the integrated *Wikifonia* and *Nottingham* collections; and in future versions, with melodic data from *Rock Corpus*, *Weimar*, and the *Band-in-a-Box* collections.

Not only does ChoCo support the creative capabilities of such systems – by providing a considerable amount of quality training data, but it also contributes to their automatic *evaluation*. In fact, the evaluation of music generation systems has recently attracted a growing interest in the field, due to the concerning ethical implications these tools are raising [370]. On one hand, this involves the extraction of statistical features quantifying the degree of alignment between a generated repertoire and the training material, with respect to certain musical properties [415]; on the other hand, it concerns the detection of potential sources

of plagiarism in generated music within and beyond the training set [419].

Another application domain that can benefit from the Chord Corpus is that of *music pedagogy*. For example, TheoryTab¹⁵ allows users to choose from a repertoire of popular songs and visualise their harmonic/melodic structure during playback – with chords encoded in both Leadsheet and Roman notations, and projected in such a way as to facilitate the theoretical understanding of the song. Chordify uses chord recognition systems to infer and align chord progressions from audio recordings, and provides support for practising them with guitar, piano, and ukulele. Despite their value, both the technology and the data powering these commercial tools are not openly available, thereby decreasing their overall wider use. In contrast, ChoCo provides an attractive open and linked solution, with its modular architecture enabling the semantic description of chords according to the desired level of complexity and granularity (e.g. an educational ontology for chords might provide a simpler vocabulary). This makes it more suitable for educational purposes.

In the context of MIR, the use of ChoCo can support a multitude of tasks. The nature of its contribution is twofold: (i) it provides an unprecedented amount of training data, which is often essential for the effectiveness of supervised methods; (ii) it contributes to the development of graph-based methodologies for music analysis that can leverage the semantic representation of chord progressions. For instance, a central research area in MIR is *music similarity*, which in turn encompasses a number of interrelated tasks, including *cover song detection* – useful for music cataloguing and to support court decisions in music plagiarism [281]; and content-based *music retrieval*, aiming to search scores or performances from musical repositories using either symbolic queries, singing (alias query-by-humming), or by playing a smart instrument [390]. Another example of a MIR task that would benefit from ChoCo is *music structure analysis* [96], which is concerned with the detection and labelling of structural segments related to musical form – a task that strongly relies on the use of harmonic/melodic features [17]. Other tasks of interest include *music tagging* [31], such as *music genre/style classification* and *composer/artist identification*. Finally, examples of tasks of musicological interest that would benefit from ChoCo include *pattern mining*, *cadence detection*, and *local key identification*.

¹⁵<https://www.hooktheory.com/theorytab>

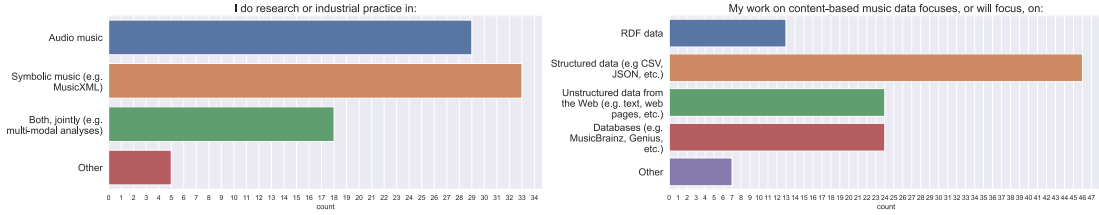


Figure 4.14: Overview of responses to Questions 2 (music domains, left), and 3 (data types, right) in the survey.

4.5.2 Online survey

Since ChoCo is a new resource for the SW, MIR and Musicology communities, we discuss here evidence for potential adoption. To gather such evidence, we performed an online survey in which we directly ask potential adopters 10 questions regarding their background, relevance, and interest in working with chord data. The online survey was distributed in the SW, International Society for MIR (ISMIR), and Digital Musicology mailing lists, gathering a total of $N = 53$ responses. The survey was conducted via Google Forms – without recording any personal data from participants or any contact information.

Results are illustrated in Figures 4.14 and 4.15. Except for questions 1-3 and 12 (multiple choices), all questions ask participants to quantify the agreement with the statement made from 1 (absolutely disagree) to 5 (absolutely agree), 3 being a neutral response (neither agree nor disagree). In the first three questions we assess the background of the respondents, finding that 38 work in MIR, 27 in Musicology, 13 in SW, and 5 are also involved in other fields (AI, Music Theory, Music Interaction). Most respondents do research or industrial practice using audio (29) or symbolic music (33), or both (18), focusing primarily on structured data when conducting content-based music studies (Figure 4.14). Nevertheless, music researchers also make extensive use of unstructured data and music databases, and 13 of them (24% of respondents), utilise RDF data.

From questions 4-11 we found that: 64% of respondents have encountered the need for chord datasets providing high-quality timed annotations of harmonic progressions, covering one or more genres/styles; 47% believe that currently existing chord datasets are not of sufficient size for their practise (whereas 41.5% have a neutral position); about 60.3% argue that such datasets do not provide content metadata sufficiently rich and informative to their needs (with another 35.8% being neutral); and 51% believe that links to external resources (e.g. MusicBrainz,

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration



Figure 4.15: Questions and overview of responses for Questions 4-11 from the online survey.

Wikidata, etc.) are rarely provided (40% are neutral). Each research community strongly recognises the value of a dataset like ChoCo as a key resource for their field: MIR (91.3%), SW (71.4%), Musicology (92.7%), and overall, 75.4% of respondents expressed their interest in using such a dataset.

4.6 Data Availability

The ChoCo dataset and Knowledge Graph, together with the ontological ecosystem and code, are publicly available from several repositories (c.f. Table 4.2). As detailed in Section 4.3.1, ChoCo is currently released in 2 modalities:

- As a JAMS dataset, where audio and score annotations are distinguished by the `type` attribute in their `Sandbox`; and temporal/metrical information is expressed in seconds (for audio) and `measure:beat` (for scores) (c.f. Section 4.3.1);
- As a Knowledge Graph, based on our JAMS ontology to model music annotations (c.f. Section 4.3.1), and on the Chord and Roman ontologies to semantically describe chords; Table 4.2 also provides links to a live SPARQL endpoint.

We have implemented a number of actions to ensure that these outputs are in compliance with the FAIR Guiding Principles for scientific data management and stewardship [406]. A GitHub repository hosts data, code, and instructions¹⁶, to fully reproduce the corpus creation from the original collections. To improve reproducibility, the repository also provides a Docker image for the project (platform agnostic). To improve data consistency, both the latest versions of ChoCo (JAMS file and RDF triples) are available on Zenodo, in synchronisation with GitHub releases.

Via GitHub and Zenodo, the ChoCo project has a unique and persistent identifier and is registered in a searchable source. Additionally, via our integration framework, ChoCo contains fine-grained provenance descriptions that allow to keep track of the original source of each harmonic annotation – both in terms of annotators (the person who contributed the harmonic analysis) and data curator (the maintainer of the original collection).

Finally, to comply with the original collections, all data and code in ChoCo is released under the *Creative Commons Attribution 4.0* licence (CC-BY 4.0), with the exception of the JAAH, CASD, and Mozart Piano Sonata subsets – which follow the *Creative Commons Attribution-NonCommercial-ShareAlike 4.0* international licence (CC-BY-NC-SA 4.0). This required an in-depth analysis of the licensing policies of the integrated collections (see Table 4.8). Indeed, for 7 collections, we could not find any specific licensing information from related scientific

¹⁶<https://github.com/smashub/choco>

Chapter 4. Harmonising Fragmented Data: A Comprehensive Workflow for Symbolic Data Integration

ChoCo subset	Original licence	ChoCo licence
Isophonics	Not specified	CC BY 4.0
JAAH	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0
Schubert-Winterreise	CC BY 3.0	CC BY 4.0
Billboard	CC0	CC BY 4.0
Chordify Annotator Subjectivity Dataset	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0
Robbie Williams	Not specified	CC BY 4.0
Uspop-2002	Not specified	CC BY 4.0
RWC-Pop	Not specified	CC BY 4.0
Real Book	Not specified	CC BY 4.0
Weimar Jazz Database	ODbL	CC BY 4.0
Wikifonia	public domain	CC BY 4.0
iReal Pro	public domain	CC BY 4.0
Band-in-a-box	Not specified	CC BY 4.0
When in Rome	CC BY-SA 3.0	CC BY 4.0
Rock Corpus	CC BY 4.0	CC BY 4.0
Mozart Piano Sonata	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0
Jazz Corpus	Not specified	CC BY 4.0
Nottingham	Not specified	CC BY 4.0

Table 4.8: *Licensing per ChoCo subset. The second column details the licence declared by the data curator of the corresponding subset; it indicates “not specified” whenever this information was not made explicit in articles, web-pages, collection metadata, repositories, etc. The last column refers to the licence attributed to the standardisation-integration output for each subset within ChoCo – which is made compliant to the original licence, as derivative work. Please, note that all the authors of the “not specified” subsets were contacted to verify whether the use of a CC-BY licence was compliant to their data publishing policies.*

articles, technical reports, online resources, repositories, dataset metadata, and so forth. For these cases, the authors of these collections were contacted and confirmed whether the use of the CC-BY 4.0 licence – on our derivative integration work – was compatible with their original releasing strategies.

4.7 Conclusion

In this chapter, we presented the *Chord Corpus (ChoCo)* as both a resource and a structured workflow for standardising and integrating symbolic datasets. This dual contribution directly addresses fragmentation issues in chord datasets by offering a scalable approach for unifying diverse digital formats and annotation practices. The proposed workflow, employed on the use case of harmonic data, guided the curation, transformation, and integration of over 20,000 high-quality

harmonic annotations from 18 different chord datasets, leveraging the JAMS data structure as a unifying annotation model.

The resulting ChoCo KG, generated through this workflow, achieves three levels of interoperability – metadata, annotation format, and chord notation – by using a standardised framework for parsing and representing diverse notational systems. This not only enhances data accessibility but also creates opportunities for advanced semantic analysis, as evidenced by the inclusion of 4,000+ links to external datasets, as well as for large-scale musicological and computational studies. Furthermore, the relevance of ChoCo was further underscored by a survey conducted with potential users from the MIR and SW communities, which demonstrated substantial interest in adopting the dataset and workflow.

For future work, we plan to expand ChoCo by incorporating additional harmonic data, further broadening its scope and application. Beyond harmonic data, the standardised workflow proposed in this chapter could be employed to produce new corpora containing various types of music annotations. In this regard, ChoCo’s workflow and the JAMS ontology have already been applied to represent a corpus of melodic pattern data [353]. Building on this approach, the workflow could be adapted to generate additional corpora focused on melodic, structural, or rhythmic annotations, enriching the resources available for diverse musicological and computational studies.

Uncovering Harmonic Similarity: From Musicological to Creative Exploration

5.1 Introduction

The creation of large, harmonized corpora of symbolic harmonic annotations opens up new opportunities for exploring harmonic data on an unprecedented scale, as discussed in Section 4.5. The integration of diverse datasets facilitates analyses across a broader range of musical genres, styles, and total number of artists and tracks in general, offering deeper insights into harmonic content that were previously hindered due to the fragmentation of the data sources.

Since the inception of MIR, similarity has been a fundamental paradigm guiding the exploration and the organisation of musical datasets. While contextual metadata plays an important role in organizing datasets and collections, content-based similarity offers several key advantages, as described in [221]: (i) it provides an objective method for comparing pieces; (ii) it does not rely on the availability of metadata, which can often be incomplete or inconsistent; (iii) it helps to overcome issues with improperly labelled pieces and ambiguous identifiers [222];

and (iv) it mitigates problems related to the long-tail in music recommendation systems, such as the “popularity bias” or the “cold-start” problems [262].

Traditionally, most content-based music similarity research has focused on audio data [109]. However, these approaches often rely on end-to-end algorithms that lack interpretability, making it difficult to understand why certain tracks are considered similar. This lack of transparency can introduce biases into similarity measures and obscure the commonalities between distinct tracks, resulting in challenges when attempting to explain the reasoning behind the system’s outcomes [231].

An alternative to audio-based methods is symbolic music similarity, which by design offers a more explainable approach. Over the past decade, symbolic similarity has been applied to various tasks, including cover song detection [101], genre classification [12], variation recognition [157], music search [84], and plagiarism detection [409]. While melodic similarity has received substantial attention, the study of harmonic similarity has not garnered as much focus in recent years. To date, state-of-the-art methods include the *TPSD* method [104] and the *CSAS* approach [175]. However, these methods often rely on global harmonic alignment, limiting their ability to detect local harmonic similarities and restricting the exploration of shared harmonic patterns among different musical works.

By utilizing symbolic harmonic annotations, this research proposes novel similarity measures that account for both global and local harmonic structures. A core objective is to demonstrate that large, unified corpora enable scalable, in-depth studies of harmonic similarity that were previously hindered by fragmented data.

In this chapter, we explore how the large-scale, harmonized corpora of symbolic annotations developed in Chapters 3 and 4 can be leveraged, alongside new similarity measures, to support both musicological exploration and music composition. In particular, we investigate how these similarity measures can empower researchers to navigate and analyse vast harmonic datasets, uncovering hidden patterns and connections across genres, musical works, and historical periods. For composers, we explore how these tools offer harmonic inspiration drawn from diverse genres and styles, sparking creativity through harmonic ideas and suggestions that may not be immediately apparent.

To achieve this, we present two main contributions: *LHARP* and *Harmory*, each implementing distinct similarity functions and tested in two different contexts—musicological analysis and assisted creativity, respectively.

5.1.1 Harmonic Similarity for Musicological Exploration

Musicological exploration enhances our ability to organize and navigate large repositories of musical data, making meaningful browsing possible for researchers and users alike. As Pampalk notes, “The value of a large music collection is limited by how efficiently a user can explore it” [297]. With access to extensive, harmonized corpora of symbolic harmonic annotations, coupled with advanced similarity measures, researchers can systematically investigate harmonic structures across genres, historical periods, and stylistic practices.

Moreover, by comparing harmonic structures at scale, we can extract underlying rules and stylistic patterns that help define specific musical genres. As Velardo observes, such large-scale analysis allows us to uncover generative patterns and evolutionary paths in music that may be challenging to identify through individual works or smaller datasets [394]. For example, the prevalence of certain chord progressions or harmonic sequences within a genre can highlight defining features of that genre.

In addition, this approach enables a quantitative study of stylistic evolution within and across genres. By tracking changes in harmonic language over time, researchers can identify shifts in genre-specific conventions and explore how certain harmonic techniques spread between genres, illustrating broader trends in musical development.

5.1.2 Supporting Music Creativity

Creativity has been defined as the ability to come up with new, surprising, and valuable ideas or artifacts [37]. These can be abstract concepts, scientific theories, solutions to real-world problems, but also new designs and artworks. As such, creativity initiates and fuels scientific discovery, knowledge creation, enables artistic expression, and on larger scale, contributes to human evolution [276]. In her seminal work, [37] categorised creativity into three types: (i) *exploratory*, where new ideas are generated by exploration of a space of concepts; (ii) *combinational*, which enables the creation of new ideas through the combination of familiar ones; and (iii) *transformational*, where the “the rules” governing a space are challenged and transformed, to generate new kinds of ideas.

A computational creativity theory was also formulated by [72], to describe creative and generative acts (FACE model) and their potential for impact (IDEA model).

Attempts at formalising human creativity date back to the ancient Greeks, and remained up to and beyond Mozart with the “*Dice Game*” and Ada Lovelace – speculating that the “calculating engine” might compose elaborate and scientific pieces of music of any degree of complexity. Since then, creativity, creative reasoning and creative problem solving have been extensively researched in cognitive [38] and computational sciences [126]. A simple definition of a computationally creative system is that of a model capable to perform “generative acts” that *create* artefacts, concepts, or provide an aesthetic *evaluation* for the generated outputs [125, 236]. By harnessing recent advancements in machine learning, a variety of systems have already been implemented across several domains. Examples include computational systems for material discovery [47], molecular design [373], and more broadly, for virtual laboratories [218]; but also models for generating textual artefacts [295], images [328, 341], and even recipes [346] from a variety of prompts.

In the music domain, data-driven generative systems based on deep learning methods have achieved impressive results on symbolic music [42], and they can also produce realistic outputs when trained on the raw audio [115]. The variety of computationally creative methods for music is quite broad and diversified, and has already enabled the exploration of novel forms of artistic co-creation [198]. These range from the automatic generation, completion, and alteration of chord progressions and melodies, to the creation of mashups, and audio snippets from textual prompts [6]. Due to their success, some of these systems have already been integrated into commercial software, such as *Aiva*¹, *Amper*², *Suno*³, and *beatoven.ai*⁴ – allowing users to generate full music pieces based on their desiderata.

Fundamental concerns of music AI systems

Nonetheless, having a system that can fully generate realistic music raises ethical concerns – especially when those systems are made commercial and can potentially replace artists, rather than augmenting their possibilities [370]. Indeed, research can open highly lucrative business opportunities given the low cost of non-human musicians and “*their inability to organise in unions to protest against unfair treatment*” [278].

¹<https://www.aiva.ai>

²<https://www.ampermusic.com>

³<https://suno.com/>

⁴<https://www.beatoven.ai/>

In addition, computationally creative models that fully learn music representations from the data by maximising a learning objective (e.g. autoregressive, masked prediction, generative modelling) are often criticised for lacking *accountability*, *explainability*, and *musical plausibility*. The former is related to the challenge of keeping track of where the model picks up while generating new musical content. As the model is unaware of its influences while composing, this may prevent giving recognition to real artists, which has direct implication on copyright and revenue sharing [123]. Similarly, the lack of explainability represents a technological barrier for users, as there is little or no understanding of the creative process underneath. Explainability is a desirable component for computationally creative systems, as it facilitates the interaction with artists, and particularly, the ability to control/steer the system based on domain knowledge [44, 39]. Finally, the “creative space” learned by data-driven systems is often criticised by musicologists and music experts in regard to musical plausibility [171], meaning that, solutions generated from these models may violate common notions of music theory. This fundamentally hampers a potential dialogue and synergies between music experts and AI researchers.

In sum, most music AI systems cannot yet be deemed trustworthy by design (accountability, explainability, ethics, etc.) [140], which raises serious concerns related to their large scale adoption.

5.1.3 Our contribution

To address these gaps, we propose two novel methods harmonic similarity methods and we employ them for diverse applications, aligning with CQ3: LHARP (Local Harmonic Agreement based on Recurring Patterns), in which we explore similarity at support of musicological analysis, and Harmory (the Harmonic Memory), where we apply similarity to explore new directions in computational creativity. These contributions, which build on the heterogeneous corpus created in Chapter 4, provide unique resources for studying harmonic relationships across genres, styles, and historical periods, supporting both academic and creative applications. These methods were detailed in two publications [93, 91].

LHARP: Local Harmonic Similarity

To overcome the limitations of global similarity measures and facilitate musicological exploration, we introduce LHARP, a new harmonic similarity function that

extends the widely used Longest Common Subsequence (LCS) into the harmonic domain. LHARP focuses on local harmonic structures, enabling the identification of recurring harmonic patterns within sequences, providing a more flexible and nuanced similarity measure. This function accommodates both exploratory and creative applications, capturing genre- and artist-specific harmonic information while maintaining generalizability across different corpora.

Additionally, we provide an interactive tool based on the induced harmonic network, allowing users to visualize and navigate harmonic relationships between chord sequences within our corpus.

Harmory: A Knowledge Graph for Harmonic Patterns

As a second contribution, we propose a novel DTW-based similarity function, and we leverage it for supporting the creation of a generative system that aims at augmenting and enhancing the creative potential of human composers, rather than replacing them [57]. Inspired by music psychology evidences [223], we present Harmory, a Knowledge Graph of harmonic patterns designed to support creative applications in a transparent, accountable, and musically plausible way.

We propose a model that leverages a cognitive model of Western tonal harmony to project chord progressions into a musically meaningful space. Using signal processing methods, we segment harmonic sequences into significant structures, which are then compared via harmonic similarity to reveal common and recurring patterns. The resulting KG establishes relationships between patterns through: (i) *Temporal links*, connecting two patterns that occur consecutively in the same progression; and (ii) *Similarity links*, connecting patterns that are highly similar in structure.

Our main contributions can be summarised as follows.

- We propose two novel *algorithms for local harmonic similarity*, designed to capture nuanced harmonic relationships and support creative applications.
- We introduce an *interactive tool* for visualizing harmonic similarities across pieces, enabling musicological exploration and discovery of harmonic relationships in large corpora.
- We contribute a novel *method for harmonic structure analysis* in the symbolic domain, leveraging cognitive and musicological models of tonal harmony.

- We release the *Harmonic Memory (Harmory)*, a large, diversified, and musically meaningful KG of harmonic patterns aimed to support applications of trustworthy machine creativity.
- We provide examples of possible applications for trustworthy machine creativity implemented on top of Harmory, focusing on knowledge discovery and human-machine chord generation.

5.1.4 Chapter Structure

This chapter is structured as follows. Section 5.2 presents the state of the art for similarity in the symbolic domain, with a particular focus on harmonic similarity, as well as computational tools designed to assist creativity. In Section 5.3, we introduce LHARP, describing both the algorithm and the interactive tool developed for exploring harmonic similarities within a subset of the ChoCo dataset. Section 5.4 provides a detailed explanation of Harmory, including the segmentation process, the novel similarity algorithms, the KG creation, and the avenues it opens for computational creativity. Finally, Section 5.5 concludes the chapter, summarizing the main contributions, their limitations and future work.

5.2 Related Work

5.2.1 Content-based Similarity in the Symbolic Domain

A significant portion of content-based similarity research in the symbolic domain focuses on *melodic similarity*. Velardo et al. [394] present a taxonomy categorising algorithms for melodic similarity detection into four main strategies: *cognition*, *music theory*, *mathematics*, and *hybrid approaches*.

Cognitive approaches emphasize human perception by using metrics and pattern recognition inspired by auditory processing, such as the combination of pitch and rhythm features [395, 340]. Music theory-based methods draw from established theoretical models, like the Generative Theory of Tonal Music (GTTM) [243], with algorithms by Grachten et al. [162] and Orio et al. [294] comparing melodic structures through annotated segments. Mathematical approaches often employ geometric and statistical methods, as in [8], where melodies are represented as polygonal chains within a pitch-time space. Hybrid systems combine various techniques from the aforementioned categories for improved accuracy and

efficiency, such as the SIMILE toolbox [144], integrating around 50 algorithms, and Fanima [375], which blends pitch and duration metrics.

Velardo’s survey also emphasizes, as a prominent issue in the field, that there is no universally accepted concept of similarity, which presents significant challenges for algorithm development and impacts the outcomes of competitive evaluations in melodic similarity detection. The lack of a shared definition similarity creates variability in algorithmic approaches and their benchmarks, as different systems may interpret and quantify similarity based on distinct criteria.

In recent years, deep learning techniques have emerged as prominent tools for deriving measures of similarity, ranging from Convolutional Neural Network (CNN) [355] to self-supervised models [320]. These approaches focus on capturing complex relationships within the data, yet the underlying concept of similarity remains ambiguous. This ambiguity is particularly pronounced given that many deep learning models are geared towards recommendation tasks [109] rather than analytical or musicological applications.

5.2.2 Harmonic Similarity in the Symbolic Domain

To the best of our knowledge, the most referential methods for harmonic similarity are the Tonal Pitch Step Distance (TPSD) [102] and the Chord Sequence Alignment System (CSAS) [175].

TPSD is a perceptually and musicologically-grounded distance function generalising Lerdahl’s Tonal Pitch Space (TPS) [242] – a model of tonality that fits musicological intuitions and correlates well with empirical findings from music cognition [99]. Given two chords, the function considers the number of steps on the circle of fifths between their roots, and the amount of overlap between the corresponding chord structures in relation to the global key. When generalised to full chord progression – an ordered sequence of chords, the TPS distance is computed between every chord and the key of the sequence. This yields a step function profiling the harmonic properties of a piece. The distance between two progressions is thus defined as the minimal area between their step functions over all possible horizontal circular shifts.

CSAS uses string matching techniques to compute similarity scores between strings representing chords or distances between chords and key. It uses the local alignment algorithm by [367] to locate and extract a pair of areas/regions from two given strings (generally defined as sequences of arbitrary symbols) that exhibit

the highest similarity with each other. Thereafter, using a dynamic programming method, a similarity score for the two progressions is calculated based on the minimum number of elementary operations (deletion, insertion or substitution of a symbol) needed to transform one sub-string into the other.

In the systematic comparison by [99], CSAS was found to perform better than TPSD on a cover detection task. Nevertheless, TPSD has stronger and more intuitive musicological interpretation, and is computationally more efficient (CSAS has quadratic time complexity).

In addition, [101] proposed a system using a generative grammar of tonal harmony to formally describe chord sequences. A parser defined from the grammar produces syntactic trees representing harmonic analyses of a given chord progression. Comparison of two different pieces is achieved by constructing a tree containing all the structures shared by their corresponding parse trees. This implements a form of tree similarity. Despite its musicological utility, the authors point out that grammar may not be expressive enough to parse certain chord sequence. In other words, chord sequence that are deemed as ungrammatical cannot be parsed, hence their similarity cannot be computed.

5.2.3 Computational Models for assisting Creativity

To the best of our knowledge, most machine learning systems are *explorative*. Starting from different prompts, such as a priming music to continue, an incomplete passage, or a textual query, these models can generate convincing outputs by sampling from the learned distribution. These include methods based on recurrent [371], self-attention [199], and convolutional neural networks [197]. Instead, current *combinational* systems are dominated by variational autoencoders, which can create new ideas by interpolating between two musical passages in a latent space [336]. *Transformative* approaches for music have been implemented by “hacking” the former methods based on the idea of brain transplant, to provide additional artistic stimulation [71]. These range from gentler interventions mixing up corpora, to splicing neural networks, jointly training with interference, and Frankensteinian hybrid models [387].

As pointed out before, most of these works lack trustworthy features to support and protect creative professionals. Recently, *Explainable Computational Creativity (XCC)* systems have been proposed, to promote a bidirectional interaction between system and user [251]. This interaction is communicative, enabling the

exchange of decisions and ideas in a format that can be understood by both humans and machines. Examples of explainable systems also include [73] – presenting a real-time human-machine interaction for artwork creation: the system provides explanations for its decisions, while users can guide the creative process.

SW technologies have also been used to make creative systems more explainable. An example is [310], which proposes a system for creating innovative food combinations using a knowledge graph that describes compounds and ingredients. However, to the best of our knowledge, no such systems have been proposed in the musical domain. A notable exception is the work by [268], enabling the generation of mashups by leveraging SW technologies for machine creativity [236]. Our work differs substantially in the broader intent and creative applications it enables, the musicological and cognitive basis, the scope/granularity of the interconnected musical content (patterns vs full pieces).

5.3 LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

Local Harmonic Agreement based on Recurring Patterns (LHARP) is a harmonic similarity function that emphasises shared repeated patterns among two arbitrary symbolic sequences. An example covering all the main steps explained below is reported in Figure 5.1 for two pop/rock pieces: “*Crazy Little Thing Called Love*” by Queen and “*P.S. I Love You*” by The Beatles.

5.3.1 Encoding of symbolic chord sequences

Before harmonic progressions can be considered for similarity, chord sequences are first pre-processed to make them comparable, and encoded in a numerical format.

The process starts with the *harmonic reduction*, where arbitrarily represented chord progression are simplified, so that only the most harmonically meaningful information is retained. For this purpose, the bass note is first discarded (e.g. $C/6$ simply becomes C). As empirically demonstrated in [99], this operation improves the generalisation capabilities of the next steps, thereby producing more consistent similarity scores. In addition, consecutively repeated chords are removed. This provides a “bird’s eye view” on the global harmonic properties of each piece. To conclude the pre-processing step, chord sequences undergo *key-based normalisation* – meaning that they are all transposed to the same key (i.e. C major).

5.3. LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

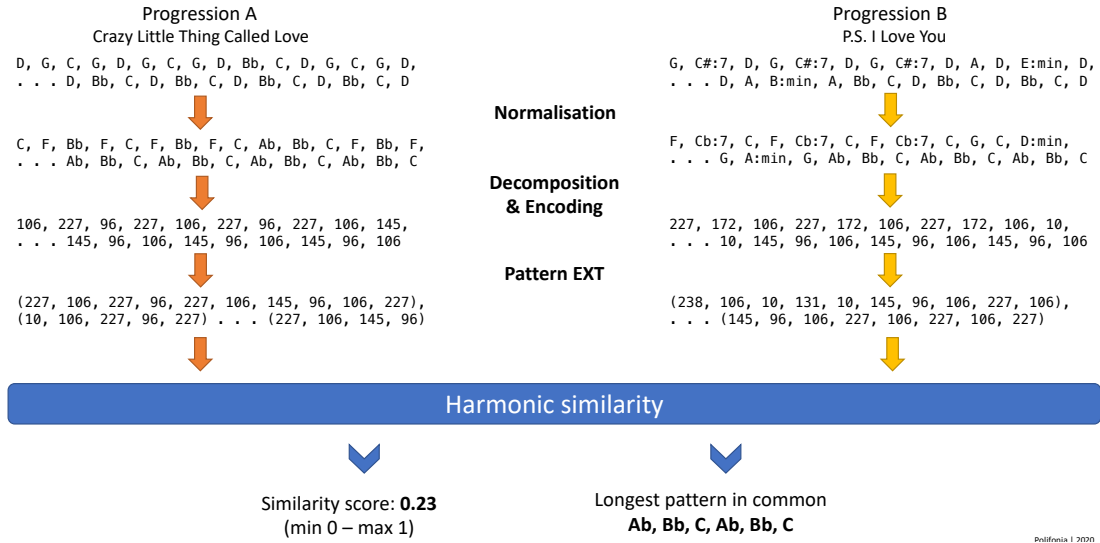


Figure 5.1: A working example of LHARP’s workflow starting from two given harmonic progressions: “Crazy Little Thing Called Love” by Queen (left) and “P.S. I Love You” by The Beatles (right).

This last step is crucial, as chords labels/classes in a progression need to be contextualised according to the key of the piece (defined by the tonic and the scale) before any comparison is possible.

The normalised harmonic sequences are then prepared for the encoding step, so that they can be used as input to any computational procedure. Rather than further simplifying the symbolic musical content, a new encoding procedure was designed to retain the fundamental internal structure of each chord. More precisely, every chord is decomposed into its pitch constituents—the individual pitches it is made of (e.g. a C major is encoded as the following set of pitches {C, E, G}). This intermediate transformation – the *decomposition of chords*, is in line with the chord encoding systems overviewed in Section 5.2.2. Finally, to reduce the complexity of any potential polyphonic model using such sparse local representations of chords, each unique decomposition is then assigned to an index (an integer value).

As it can be observed, this approach – the *enumeration of pitch simultaneities*, is akin to the common encoding methods used in natural language processing for word tokens. Nonetheless, enumerating the decomposition of chords rather than their actual labels, is expected to drastically reduce the vocabulary size. Indeed, if two distinct chord labels have the same decompositions, they will be associated to the same token/index. In sum, for each harmonic progression, the output of this

last step is a sequence of *chord tokens* defined over the vocabulary of all possible chord decompositions.

5.3.2 Pattern extraction and matching

Once two given chord sequences have been encoded, the next step is to extract all their internal repetitions and compute a similarity score based on their shared structures.

The first step is carried out independently on each chord sequence. To identify the regions of chord progressions that can be expected to be “harmonically memorable”, we extract the n -grams of all possible orders – starting from $n \geq 3$ (i.e. from tri-grams), that repeat at least once within the progression. The resulting set of “harmonic thumbnails”, representing potentially memorable harmonic structures in each progression, is denoted as *Bag of Recurring Patterns (BRPs)*.

Chord progressions are then compared for similarity based on the agreement between their BRPs. In particular, the longest harmonic structures they share is compared to the order of the longest thumbnail that occurs within each progression, independently. Depending on the harmonic patterns the two chord progressions have in common – in relation to their internal structures, the similarity function will return a value between 0 and 1 (the higher the value, the stronger the similarity), together with the longest harmonic patterns they share. For example, if the longest harmonic thumbnail two chord sequences share has order 5, whereas the longest recurring patterns in their BRPs has order 10, the similarity function will return a value of 0.5.

Formalisation

For a more detailed understanding of the similarity function, here we provide a generalised formalisation, which applies to any uni-modal symbolic sequence. To explain our method, we first introduce some basic notation and definitions. Let \mathbf{C} denote a sequence of tokens drawn from a vocabulary V and belonging to a dataset of chord sequences $\mathcal{D} = \{\mathbf{C}_1, \dots, \mathbf{C}_l\}$, defined as:

$$\mathbf{C} \in \mathbb{Z}^{m \times n} \text{ s.t. } m = |V|, n = \max(\{|\mathbf{C}| \text{ for } \mathbf{C} \in \mathcal{D}\}) \quad (5.1)$$

$$\mathbf{C} = \mathbf{c}^{(1)}, \mathbf{c}^{(2)}, \dots, \mathbf{c}^{(n)} \text{ s.t. } \mathbf{c}^{(i)} \neq \mathbf{c}^{(i+1)} \forall i \in [1, n] \quad (5.2)$$

5.3. LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

where l is the size of the dataset (the number of chord sequences it provides) and n corresponds to the size of the longest chord sequence in \mathcal{D} . This allows to homogeneously represent the dataset as a tensor $\mathcal{D} \in \mathbb{Z}^{l \times m \times n}$. With this notation, we now define the *recurring patterns* (or *harmonic thumbnails*) $p_k(\mathbf{C})$ of order k for the sequence \mathbf{C} as follows:

$$p_k(\mathbf{C}) = \{(\mathbf{c}^{(i)}, \mathbf{c}^{(i+1)}, \dots, \mathbf{c}^{(k)}) \mid \exists j \neq i \text{ s.t. } (\mathbf{c}^{(j)}, \mathbf{c}^{(j+1)}, \dots, \mathbf{c}^{(k)}) = (\mathbf{c}^{(i)}, \mathbf{c}^{(i+1)}, \dots, \mathbf{c}^{(k)})\}. \quad (5.3)$$

Note that from this formulation we are assuming that an n -gram will be considered a *recurring pattern* only if it repeats at least once in the sequence. Nevertheless, this assumption can be easily parametrised if more control is needed (e.g. we can require more repetitions to occur in order to characterise a pattern). An example of *bi-gram recurring pattern* is given below.

$$p_2(\mathbf{C}) = \{(\mathbf{c}^{(i)}, \mathbf{c}^{(i+1)}) \mid \exists j \neq i \text{ s.t. } (\mathbf{c}^{(j)}, \mathbf{c}^{(j+1)}) = (\mathbf{c}^{(i)}, \mathbf{c}^{(i+1)})\}$$

For a sequence \mathbf{C} there can be an arbitrary number of recurring patterns of any possible order k , but the maximum order will trivially be $\hat{n} = \frac{n}{2}$ in the extreme case that the second half of \mathbf{C} fully repeats the first half. Therefore, we define the *bag of recurring patterns* $P(\mathbf{C})$ that can be extracted in \mathbf{C} as follows:

$$P(\mathbf{C}) = \bigcup_{k=K}^{\hat{n}} p(\mathbf{C}) \quad (5.4)$$

where $K \leq \hat{n}$ is a hyper-parameter expressing the minimum order of the recurring patterns. This function can be particularly expensive to compute, as the complexity is at least quadratic for each n -gram to extract. However, extracting the recurring patterns of order k already “repeats a lot of computation” for the more granular orders ($< k$). This consideration can thus be used to speed up computation and mitigate the complexity of this procedure.

The bag of recurring patterns $P(\mathbf{C})$ is thus defined as the set containing all the recurring patterns of $k \geq K$ order in the given sequence, and forms the basis of our similarity function. For convenience, we define a function measuring the degree of maximal repetition.

$$d(P(\mathbf{C})) = \max(k) \text{ s.t. } \exists p_k \in P(\mathbf{C}). \quad (5.5)$$

From these concepts, we can finally define our measure of similarity between two sequences **A** and **B** (both analogous to **C** in our notation above) as follows.

Similarity ranges in $[0, 1]$ and is minimum when **A** and **B** do not have any recurring pattern in common; it is maximum when the the longest recurring pattern in common has the same length to both the longest patterns that appear internally in each sequence ($d_{A,B} = d(P(\mathbf{A})) = d(P(\mathbf{B}))$).

A limitation of this formulation is its inability to handle cases where one or more sequences lack internal repetitions. This also includes the situation in which a pattern appears repeated only across the sequences, rather than in each of them in the first place. An extreme case is when the sequences are identical but have no patterns:

$$\mathbf{A} = \mathbf{B} \wedge P(\mathbf{A}) = P(\mathbf{B}) = \{ \}.$$

An example of this particular case could be a *rondo form*, where the degree of internal repetition could be little or none. To account for this, at the expense of computational complexity, we define a correction term:

$$h_{cross} = \frac{d(P(\mathbf{A} \oplus \mathbf{B}) - P(\mathbf{A}) - P(\mathbf{B}))}{\min(|\mathbf{A}|, |\mathbf{B}|)} \quad (5.6)$$

where the $\mathbf{A} \oplus \mathbf{B}$ denotes the concatenation of the corresponding sequences – as if they describe the same process. This allows to detect patterns where **A** repeats something from **B** or **B** repeats something from **A**, but these n-grams are never repeated in **A** and/or in **B**. Our next step is to combine h_{sim} and h_{cross} within the same formula, which can be simply done as follows.

$$h_{global}(\mathbf{A}, \mathbf{B}) = \max(h_{sim}(\mathbf{A}, \mathbf{B}), h_{cross}(\mathbf{A}, \mathbf{B})) \quad (5.7)$$

5.3.3 Preliminary experiments

To evaluate LHARP as a method for harmonic similarity, we perform a graph analysis methods to encode harmonic dependencies (edges) between music pieces (nodes) based on their similarity values. This was done to test whether network structures that can be statistically attributed to different genres/styles emerge from the network.

For this experiment, all similarities were computed using the first term of the h_{global} function in Equation 5.7 – meaning that only h_{sim} is retained for extracting

5.3. LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

similarity scores. This last decision was motivated by type of music contained in our dataset, which already exhibits a great deal of internal repetition. Moreover, the minimum order of recurring patterns was set to trigrams ($K = 3$ in Equation 5.4), to avoid inflating the number of resulting harmonic dependencies from potentially uninformative structures (bi-grams).

The dataset used for this analysis was a subset of ChoCo 4, specifically three representative partitions: Isophonics [258], JAAH [129], and Schubert Winterreise [402]. These partitions collectively represent the range of musical genres within ChoCo, providing a balanced sample for evaluating harmonic similarity across different styles.

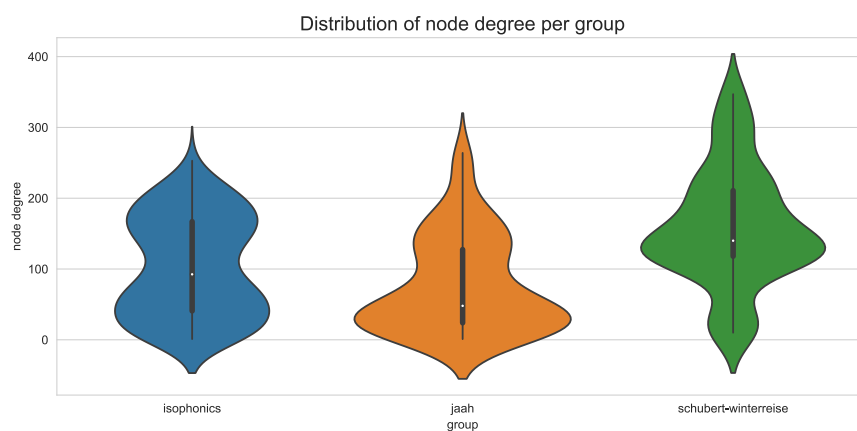
In total, 525 tracks were used in the experiment. Although this dataset size is relatively modest, it supports effective musicological exploration and simplifies the proposed graph exploration in Section 5.3.4, allowing for clear insights into genre-specific harmonic dependencies and facilitating focused analysis on network structure.

5.3.4 Analysis of genre-specific harmonic dependencies

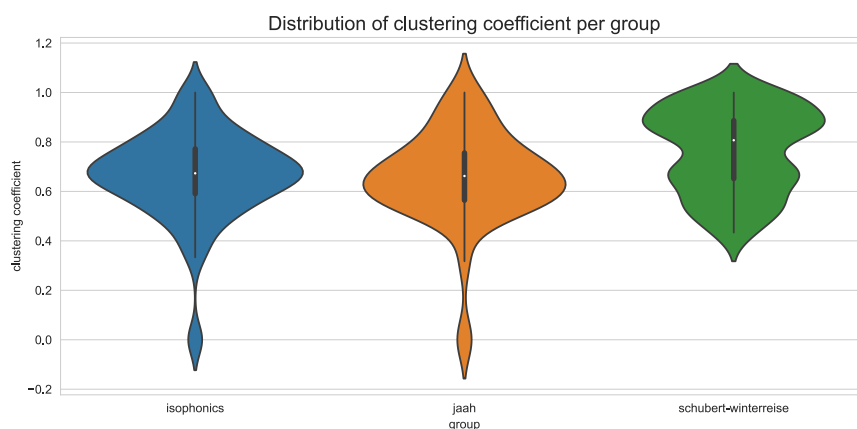
The goal of this experiment is to test whether LHARP can capture genre-specific harmonic properties when computed across different datasets. In particular, after representing musical pieces as nodes and non-negative harmonic similarities (among pieces) as edges, our hypothesis is that the graph structures emerging from the induced network (node properties, clusters, etc.) already encode genre information. This is done in two different ways: (i) by computing a set of node-specific metrics encoding structural properties of nodes, and analysing their distribution for each dataset/group; (ii) by performing community detection on the graph to verify if any alignment between network communities/clusters and (genre-specific) chord datasets can be found. Both these methods are of unsupervised nature.

Common to both studies is the creation of the *harmonic network* – a graph representing harmonic dependencies among tracks. Formally, the graph is defined as $G = (V, E)$, where V denotes the set of all chord progressions (identified by the name of each track) as nodes, and E is the set of edges connecting pairs of nodes if and only if their harmonic similarity is strictly positive. The weight of each edge corresponds to the similarity between the associated tracks, thus ranging in $(0, 1]$. Since harmonic similarities are symmetric, the resulting network is an undirected graph with no self-loops and at most one edge between any two nodes.

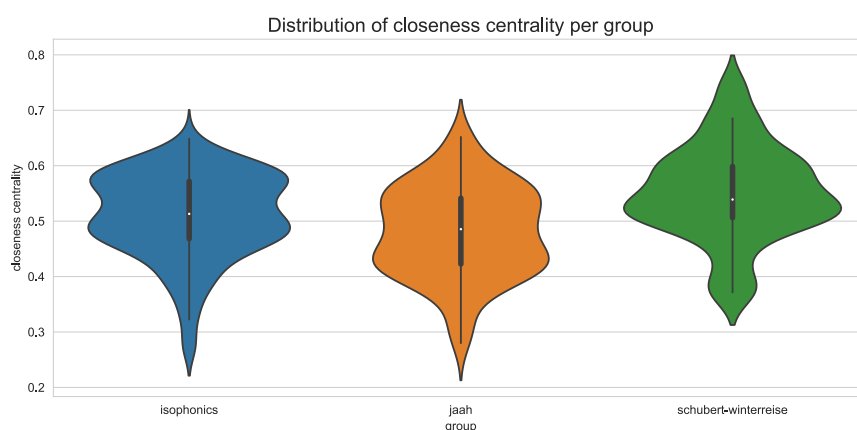
Chapter 5. Uncovering Harmonic Similarity: From Musicological to Creative Exploration



(a) *Degree*



(b) *Clustering Coefficient*



(c) *Closeness Centrality*

Figure 5.2: Distribution of each group/dataset on the three node-specific network metrics: degree (a), clustering coefficient (b), and closeness centrality (c). This is accompanied with the results of the post-hoc statistical analysis in Table 5.1.

5.3. LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

The harmonic network for this experiment is constructed by computing LHARP (c.f. Section 5.3) on each pair of chord progressions from the three datasets outlined in Section 5.3.3. This results in an adjacency matrix encoding the non-null pair-wise harmonic similarities among all tracks in the datasets; in sum, the output of this process is a suitable representation of E .

Statistical analysis of network metrics

From the resulting graph, the following metrics were computed for each node: *degree*, *clustering coefficient*, and *closeness centrality*. The degree of a node $v \in V$ is defined as the number of edges adjacent to v . Given that the graph is undirected, with one edge at max per node couple, this corresponds to the number of tracks sharing harmonic similarities with v . The clustering coefficient of v is the fraction of possible triangles through v that exist – a measure expressing the propensity of a node to form cluster with its neighbours. Finally, the closeness centrality of v is formulated as the reciprocal of the average shortest path distance to v over all the other reachable nodes in the graph [143].

To analyse if genre-specific network properties exist, each metric was studied independently with respect to the dataset each node/track belongs to. The resulting distributions are illustrated in Figure 5.2. From the results of a Kruskal-Wallis H-test, we found that the distributions of each metric associated to the three datasets differ significantly ($\chi^2 = 67.43$, $\chi^2 = 34.13$, $\chi^2 = 44.92$ for node degree, clustering coefficient, and closeness centrality respectively), with p-value less than 0.0001. Post-hoc multiple comparisons (Kolmogorov-Smirnov tests) were then performed to detect significant differences between each pair of groups/datasets in relation to each metric (Bonferroni corrections were applied to account for multiple comparisons). The results of this analysis are reported in Table 5.1, demonstrating that all these groups are statistically different from each other, with the exception of the clustering coefficient for Isophonics and JAAH.

	degreedegree			clustering coefficientclustering coefficient			closeness centralitycloseness centrality		
	isophonicsisophonics	jaahjaah	schubertschubert	isophonicsisophonics	jaahjaah	schubertschubert	isophonicsisophonics	jaahjaah	schubertschubert
isophonicsisophonics	—	****	**	—	nsns	***	—	****	***
jaahjaah	*****	—	***	nsns	—	****	—	—	***
schubertschubert	*****	****	_	*****	*****	_	*****	*****	_

Table 5.1: Summary of the Kolmogorov-Smirnov tests used to detect statistically significant differences between the groups/datasets on each metric. Conventions: *ns* denotes non-significance ($p \geq 0.05$); * denotes $p \leq 0.05$; ** $p \leq 0.01$; *** $p \leq 0.001$.

Tracing genres as network communities

To complement our analysis, we used community detection methods to assess whether nodes/tracks can be clustered into genre-specific clusters directly from their harmonic relationships. A number of algorithms were cross-validated on the graph, based on two unsupervised clustering descriptors measuring the quality of a partition: *coverage* and *performance*. The former is the ratio of intra-community edges to the total number of edges in the graph, whereas the latter adds the inter-community edges to the ratio, now in relation to the total number of potential edges. The community detection method that maximised these metrics on the harmonic network was the Fluid Communities algorithm [299]. As illustrated in Figure 5.3, we found a consistent overlap between communities and datasets: approximately 80% of the nodes in Isophonics (pop/rock music) falls within the C0 community, 70% of JAAH (jazz music) in C1, and 90% of Schubert-Winterreise (classical music) in C3. Therefore, we have enough evidence to conclude that the structural properties of the harmonic network encodes genre-specific information.

5.3.5 The interactive harmonic network

We designed a Web interface allowing users to inspect the harmonic network and interact with its components for a granular control. Figure 5.4 reports the main visualisation panel of the tool for the harmonic network constructed from the three chord datasets in our first experiment (Section 5.3.4). An online version of the interactive tool is available at <https://polifonia-project.github.io/musilar-preview/> with a video demo at https://youtu.be/NW_9z_fL7uI.

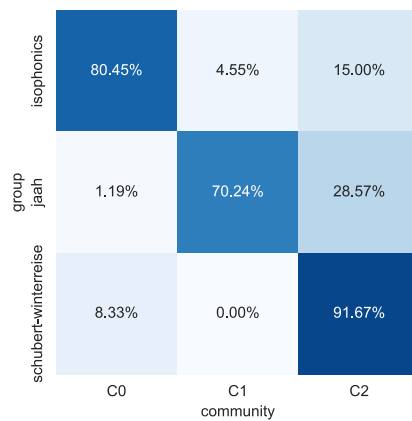


Figure 5.3: Illustration of how node/tracks are spread across the three communities.

5.3. LHARP: A Local Harmonic Similarity Function Based on Shared Repeated Chord Structures

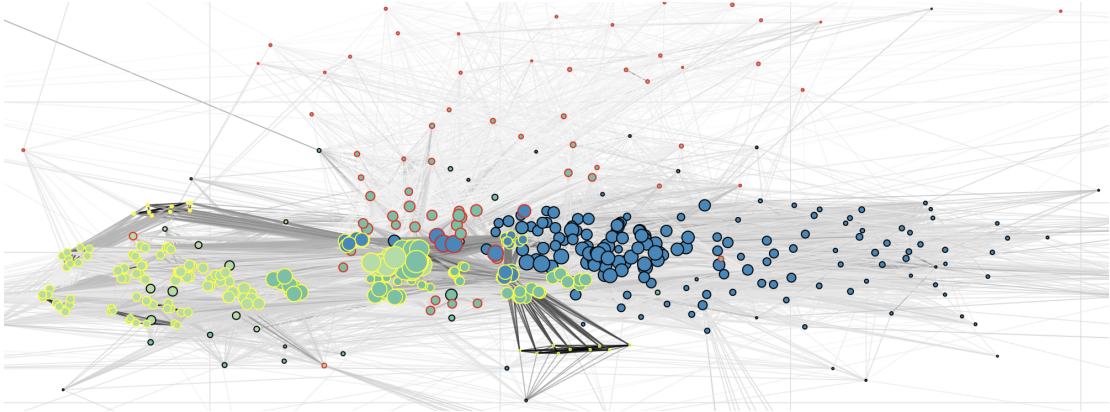


Figure 5.4: *Illustration of the harmonic graph, encoding all the harmonic dependencies between tracks in the three music collections, using LHARP.*

To facilitate the exploration of the network, a grey-scale colourmap visualises the harmonic similarity expressed by the edge weights: from light grey (low similarity) to plain black (high similarity). Nodes are sized according to their degree, whereas their border colour uniquely identifies the music dataset they belong to. In this case, a yellow border is used for nodes/tracks in the Schubert-Winterreise dataset, a black contour for Isophonics, and red for JAAH.

To interact with the network, users can first zoom in/out the main panel – focusing on specific regions of the graph, and display information about nodes and edges. By hovering over nodes they can visualise the title and artist of the corresponding track, together with its degree and community class (c.f. Section 5.3.4). Analogously, hovering over edges shows the value of harmonic similarity between the corresponding tracks, and the longest recurring pattern they have in common. Another feature useful for the inspection of highly connected regions is the node selection, isolating a node/track in such a way as to emphasise all the connections (harmonic matches) associated to it. Finally, a slider allows to perform a dynamic filtration of the graph, by discarding all edges with harmonic similarity outside a predefined range.

Overall, the tool provides a computational infrastructure to interpret the results of our experiments, and also to perform validation studies for music similarity methods. In addition, the interface can be used to discover new relationships among artists, composers, tracks, but also to test musicological hypotheses. For instance, users may discover that two authors use similar but not identical harmonic structures, even though there is no direct and strong connections between them, but possibly through the influence of a third entity. The harmonic network

also offers a valuable resource for educational purposes, providing an engaging tested for teaching network analysis concepts using music as an applied domain. For instance, MSc Computer Science students at King’s College London have utilised the harmonic network within their Network Data Analysis module. This real-world dataset allowed students to apply and test their understanding of graph analysis metrics, as well as model musical influence through epidemic models, enhancing their grasp of theoretical concepts through practical application.

5.4 Harmony: The Harmonic Memory

The main steps for the creation of Harmony are illustrated in Figure 5.5, and encompass four stages: (i) projection of harmonic sequences in the Tonal Pitch Space; (ii) novelty-based segmentation of harmonic sequences; (iii) pattern identification through similarity-based linking of harmonic segments; and (iii) KG creation.

Our workflow is defined from the harmonic analysis of a piece, which contains a sequence of *chords* in Harte notation [181], their *onsets*, and the associated local *keys*. Formally, let $\mathbf{c} = \{c_1, \dots, c_N\}$ denote a chord sequence of length N , where each chord figure c_i is drawn from the Harte chordal set \mathcal{H} . Similarly, $\mathbf{k} = \{k_1, \dots, k_N\}$ denotes the corresponding local keys of \mathbf{c} , s.t. each k_i is a tonic-mode tuple defined from $\mathbb{T} \times \mathbb{M}$, where $\mathbb{T} = \{Ab, A, A\#, Bb, \dots, G\#\}$ is the set of all possible tonic notes, and $\mathbb{M} = \{\text{major, dorian}, \dots, \text{locrian}\}$ is the set of all possible modes in Western tonal music.

For simplicity, chords are expected to be temporally aligned with their onsets, meaning that c_i ends when c_{i+1} starts, $\forall i \in N - 1$. Hence, *onsets* are defined

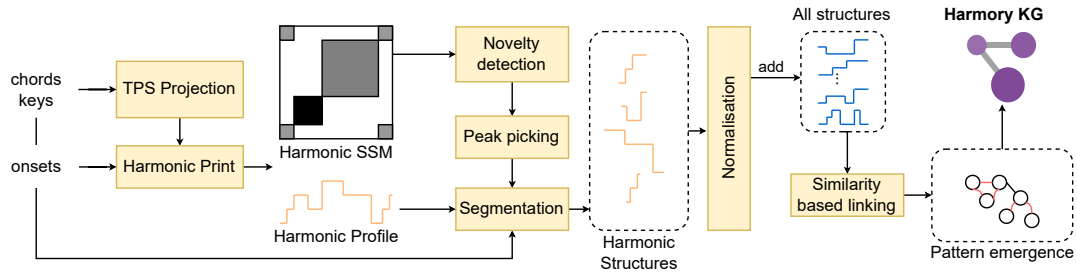


Figure 5.5: Overview of the main steps for the creation of Harmony, from the encoding of chord progressions in the Tonal Pitch Space (TPS) and their segmentation, to the emergence of harmonic patterns through similarity and the creation of the KG.

as a $(N + 1)$ -th dimensional vector $\mathbf{t} \in \mathbb{R}^{N+1}$ to compensate for the end time of the last chord (t_{N+1} is the end of c_N). Onsets are given in seconds for harmonic analyses on audio music; or as global beats for symbolic music. For example, $\mathbf{c} = [\text{G}, \text{B:min}, \text{E:min7}, \dots]$, $\mathbf{k} = [(\text{G}, \text{major}), (\text{G}, \text{major}), (\text{G}, \text{major}), \dots]$, and $\mathbf{t} = [1, 3, 5, \dots]$ are the first three occurrences of such vectors for a “*A Day in the Life*” by The Beatles.

Encoding chords in the Tonal Pitch Space

Given a harmonic analysis $\mathbf{H} = \{\mathbf{c}, \mathbf{k}, \mathbf{t}\}$, the first step is to encode \mathbf{c} and \mathbf{k} as a numerical stream, so as to allow the processing of similarity/distance operations. This is necessary because chords (\mathbf{c}) and tonalities (\mathbf{k}) are complex elements to process, and come in symbolic format. More specifically, a chord label is a convention for describing intervals built on a root note. For example, the label of a C major seventh chord (C:maj7) represents the intervals of a major quadriad with a minor seventh built on the note C , which is equivalent to the note set $\{C, E, G, Bb\}$. Also, the harmonic function of a chord is contextual to the global (and local) key [5].

One option here is to leverage Representation Learning methods on symbolic music to learn harmonic embeddings from a large corpus of chord sequences [234, 237]. These include static embedding methods, such as Word2Vec [275] and Glove [304], as well as sequence models for contextualised representations, such as ELMo [307] and BERT [114] – which have proved their efficacy on a variety of natural language processing tasks. Nonetheless, in the music domain, representation learning methods have recently started to gain success for audio music [217], whereas little attention has been given to symbolic music. This is exacerbated by the challenge of finding musicological interpretability of the resulting embeddings, requiring new probing and evaluation methods for music [171].

We aim for an encoding of harmony that is well established, perceptually and musicologically plausible, and explainable by design. Hence, we rely on the Tonal Pitch Space [242] – a cognitive model of tonality used in music psychology and computational musicology.

The tonal pitch space

The *Tonal Pitch Space* (TPS) model [242] provides a scoring mechanism that predicts the proximity between musical chords. It is based on the Generative

Theory of Tonal Music [243] and designed to make explicit music theoretical and cognitive intuitions about tonal organisation. The model works by comparing any possible chord to an arbitrary key, by means of the *basic space*. The basic space is constituted by five different levels, ordered from the most stable to the least stable: (i) the *Root level*; (ii) the *Fifths level*; (iii) the *Triadic level*; (iv) the *Diatonic level*; and (v) the *Chromatic level*.

Each level holds one or more notes, indexed from 0 (*C*) to 11 (*B*). The *Root* level holds the root of a chord (0 for C-major), while the *Fifths* level adds the fifth (0, 7 for C-major). The *Triadic* level has all the notes in the chord (0, 4, 7 for C major). The *Diatonic* level depends on the chord’s key as it holds all the notes of the diatonic scale of the key (0, 2, 4, 5, 7, 9, 11 for the C major key). Finally, the *Chromatic* level holds the chromatic scale (0-11).

The distance between two chords c_i, c_j in keys k_i, k_j is calculated using the *basic spaces* of the chords. The basic space is set to match the key of the pieces (level *iv*), and their levels (*i-iii*) are adapted to match the chords to be compared. The Chord distance rule is applied to calculate the distance. The Chord distance rule is defined as $d(x, y) = j + k$, where $d(x, y)$ is the distance between chord x and chord y ; j is the minimum number of Circle-of-Fifths rule applications to shift x into y , and k is the number of non-common pitch classes divided by 2 in the levels (*i-iv*) of the basic spaces of x and y . The Circle-of-Fifths rule consists in moving the levels (*i-iii*) four steps to the right or left on level *iv*.

For each comparison between two chords, the TPS returns a value in $[0, 13]$. TPS has been demonstrated to be sound both musicologically and perceptually [103, 104], and in this work, it is used to encode and compare chord-key pairs.

Novelty-based harmonic segmentation

The projection of chord-key pairs (c_i, k_i) in the TPS is a fundamental requirement to perform harmonic segmentation. First, the given harmonic annotation \mathbf{H} is used to sample a signal \mathbf{X} of length $d = t_{N+1} \cdot f_s$, where harmonic observation (c_i, k_i) is consecutively repeated $t_i \cdot f_s$ times (its duration), according to a sampling rate f_s . Each element $x_i \in \mathbf{X}$ now encodes an input for the TPS model, containing the harmonic content at the i -th sample.

The resulting signal allows for the computation of two harmonic descriptors, i.e., the *Harmonic Profile* (or TPS time series), and the *Harmonic Self Similarity Matrix* (*SSM*) – the entry point for segmentation. The former is defined as a

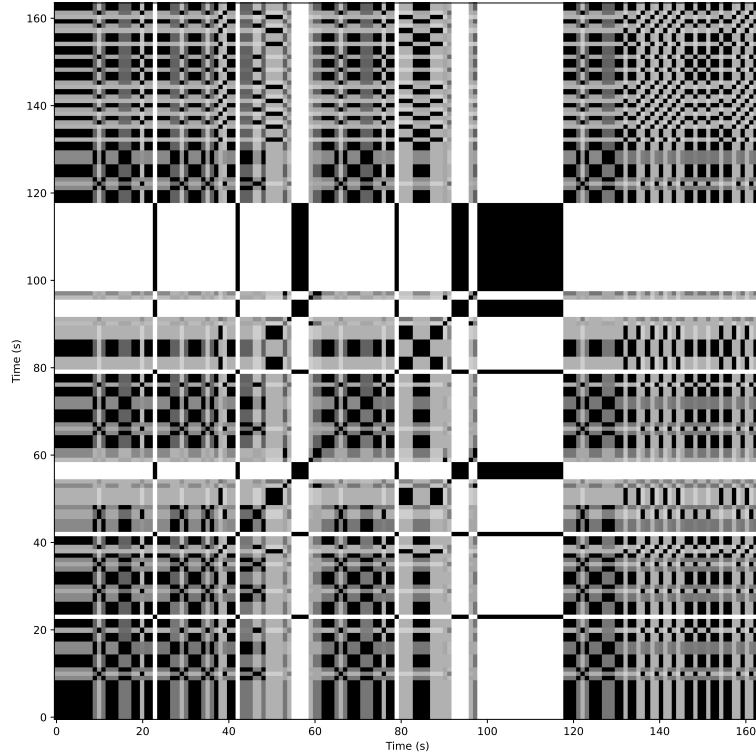


Figure 5.6: *Example of Harmonic SSM resulting from the application of Equation 5.8 on the TPS signal of “Crazy Little Things Called Love” by Queen, using a sampling rate $f_s = 1$. Four main block-like structures are visible, correlating with the musical form of the piece. Smaller, nested harmonic structures of lower granularity are observed within these blocks.*

vector $\mathbf{q} \in \mathbb{R}^d$ s.t. $q_i = \text{tps}(x_i, k_1)$, holding the TPS distance between each harmonic observation x_i and the global key k_1 of the piece (assumed as the first key occurrence). Similarly, the Harmonic SSM is a matrix $\mathbf{S} \in \mathbb{R}^{d \times d}$ s.t.

$$\mathbf{S}(n, m) = 1 - \frac{\text{tps}(x_n, x_m)}{13}, \quad (5.8)$$

where $x_i \in \mathbf{X}$ is a column vector; $n, m \in [0 : d-1]$; 13 is a normalisation factor (the maximum TPS value); and the subtraction from 1 is used to obtain a similarity score from a distance measure.

Self-similarity matrices have been extensively used for structure analysis, due to their ability to reveal nested structural elements [141, 96]. As can be seen from Figure 5.6, block-like structures are observed when the underlying sequence shows homogeneous features over the duration of the corresponding segment. Often, such a homogeneous segment is followed by another homogeneous segment that stands

in contrast to the previous one.

To identify the boundary between two homogeneous but contrasting segments (2D corner points), we slide a checkerboard kernel \mathbf{K} along the main diagonal of the SSM and sum up the element-wise product of \mathbf{K} and \mathbf{S} . A checkerboard kernel can be simply defined as a box kernel $\mathbf{K}_B \in \mathbb{Z}^{M \times M}$ where $M = 2L + 1$ is the size of the kernel, defined by $\mathbf{K}_B = \text{sgn}(k) \cdot \text{sgn}(l) \forall k, l \in [-L, L]$, where sgn is the sign function. This yields a novelty function $\Delta_{\text{kernel}}(n)$ for each index $n \in [1 : d]$ of \mathbf{X} as follows:

$$\Delta_{\text{kernel}}(n) = \sum_{k, l \in [-L, L]} \mathbf{K}(k, l) \cdot \mathbf{S}(n + k, n + l) \quad (5.9)$$

for $n \in [L + 1 : d - L]$. When \mathbf{K} is located within a relatively uniform region of \mathbf{S} , the positive and negative values of the product tend to sum to zero (small novelty). Conversely, when \mathbf{K} is at the crux of a checkerboard-like structure of \mathbf{S} , the values of the product are all positive and sum up to a large value (high novelty) [282].

Local maxima of the novelty curve are then used to detect the boundaries of neighbouring segments that correspond to contrasting harmonic parts. For this, we use a pick peaking method that applies a smoothing filter to the novelty function (to reduce the effect of noise-like fluctuations) and uses adaptive thresholding to select a peak when novelty exceeds a local average [291]. The detected segment boundaries are used to split \mathbf{X} and the corresponding \mathbf{q} into a number of non-overlapping harmonic structures. This yields $\bar{\mathbf{q}} = \bar{\mathbf{q}}^1, \dots, \bar{\mathbf{q}}^P$, where P denotes the number of structures.

Linking harmonic segments via similarity

Each harmonic structure $\bar{\mathbf{q}}^i$ is then considered for harmonic similarity. Since $\bar{\mathbf{q}}^i$ is still a time series (a partition of \mathbf{q} , the Harmonic Profile), we formulate the harmonic similarity between two harmonic structures by comparing their time series. This is done using Dynamic Time Warping (DTW) – an algorithm for comparing time series, which has been widely used across various domains, including speech recognition [280], pattern recognition [348], and bioinformatics [420]. In our case, DTW has desirable properties, as it is invariant to time shifts, and robust to local variations.

Vanilla DTW compares two time series by calculating the cumulative distances

between each point/observation. It allows for non-linear alignment between the time series by considering the local warping path. The cost matrix, holding the cumulative distance between each corresponding point, is constructed using the Euclidean distance, and is formalised as:

$$d_{DTW}(\bar{\mathbf{q}}^i, \bar{\mathbf{p}}^j) = \sqrt{\sum_{(v,w) \in \pi} \|\bar{q}_v^i + \bar{p}_w^j\|^2} \quad (5.10)$$

where π is the optimal warping path – the shortest cumulative distance between the time series (found via dynamic programming).

As the computational complexity of vanilla DTW is quadratic in the sequence length, here we use the *Sakoe-Chiba* variant. The latter achieves linear complexity $\mathcal{O}(N \cdot w)$, by constraining the warping path within a window of size w , rather than using all points (N).

Prior to the computation of similarities, time series are normalised and resampled to meet the same length, and standardised to zero mean and unit variance. This has the effect of comparing time series by looking at their shapes in an amplitude-invariant manner – which brings us closer to the identification of harmonic patterns.

The latter emerge after retrieving the k most similar structures for each segment $\bar{\mathbf{q}}^i$, and applying a filter to retain only those structures $\bar{\mathbf{p}}^i$ whose $d_{DTW}(\bar{\mathbf{q}}^i, \bar{\mathbf{p}}^j)$ is below a given threshold. Structures sharing the same (normalised) TPS time series ($d_{DTW} = 0$) define a distinct harmonic pattern; whereas segments with similar time series can be grouped within the same pattern family/cluster.

5.4.1 Knowledge graph creation

An ontology, called *Harmory Ontology*, was developed for the creation of the KG. The ontology re-uses the Core module from the *Polifonia Ontology Network (PON)* [88] (see Chapter 3). This allows to link Harmory to ChoCo⁵ [95]. We also align to the *Music Ontology* [324] – a widely used ontology model in the music domain.

For each piece, the ontology allows to: (i) store its metadata, such as title, genre, and artist; (ii) hold the harmonic segmentation (see Section 5.4); and (iii) relate similar segments (see Section 5.4). This enables semantic access to the aforementioned data via SPARQL.

⁵ChoCo SPARQL endpoint: <https://polifonia.disi.unibo.it/choco/sparql>

Chapter 5. Uncovering Harmonic Similarity: From Musicological to Creative Exploration

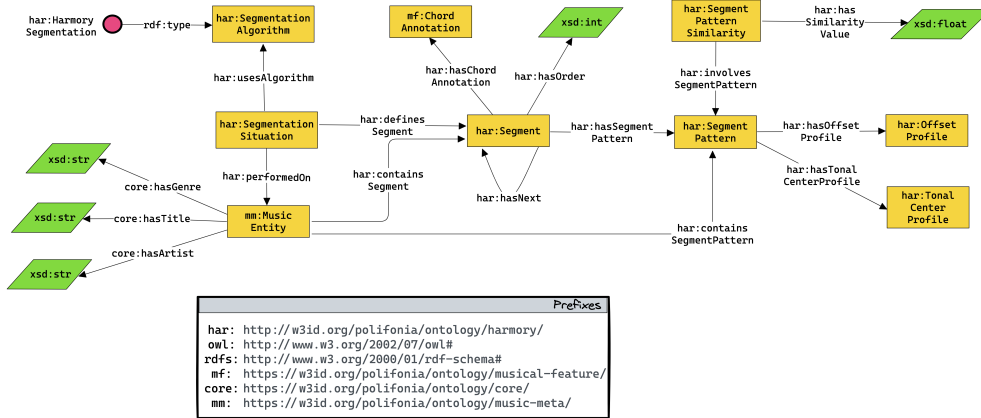


Figure 5.7: Graffoo diagram illustrating the Harmony Ontology.

The model is illustrated in Figure 5.7, using the Graffoo notation⁶. A piece of music is described by means of the class `mm:MusicEntity` which is imported from the Music Meta module of PON. Leveraging this core class from the Music Meta module enables seamless linkage of the music track—and consequently its similarity data—with the extensive range of information that can be described across all PON modules.

A musical work has a `har:SegmentationSituation` – a specialisation of the *Situation Pattern* [149] describing a segmentation performed by a specific `har:SegmentationAlgorithm` that produces one or more `har:Segments`. In this context, a harmonic sequence is split/partitioned into a number of segments, with their ordering allowing for reconstruction. Each sequence also holds its chords, using the class `mf:Chord`. Each segment is linked to its corresponding `har:SegmentPattern` – an abstraction of the TPS pattern normalised on the temporal axis. Hence, several `har:Patterns` may have the same `har:SegmentPattern`. Similarity relations are expressed via the class `har:SegmentPatternSimilarity`, which relates two Segment Patterns and holds their similarity value via the datatype property `har:hasSimilarityValue`.

5.4.2 Experiments

To validate Harmory, we tested the efficacy of the two central components underpinning its creation: the DTW harmonic similarity (Section 5.4), and the harmonic segmentation (Section 5.4).

⁶Graffo Notation: <https://essepuntato.it/graffoo/>

Evaluation of harmonic similarity

We evaluated the DTW harmonic similarity by comparing our implementation with other algorithms for the *cover song detection* task – a common benchmark for similarity algorithms in the symbolic music domain [103, 104].

In this comparison, performance is evaluated using two standard metrics: *First Tier* and *Second Tier*. The former measures the ratio of correctly retrieved songs within the top $(C_t - 1)$ matches to $(C_t - 1)$, where C_t is the size of the song class (e.g. the same composition, or performance) for track t . The First Tier can be formalised as:

$$FirstTier_{(D)} = \frac{1}{N} \sum_{t=0}^N \frac{||L_{|(C_t-1)|} \cap C_t||}{||(C_t - 1)||}, \quad (5.11)$$

where N is the set of all tracks in the dataset having at least a “cover”, and $L_{(C_t-1)}$ denotes the list of matches for track t ranked by similarity – where only the first $(C_t - 1)$ occurrences are considered. Similarly, the Second Tier is defined as the ratio of correctly retrieved songs within the best $(2C_t - 1)$ matches to $(C_t - 1)$.

$$SecondTier_{(D)} = \frac{1}{N} \sum_{t=0}^N \frac{||L_{|(2C_t-1)|} \cap C_t||}{||(C_t - 1)||} \quad (5.12)$$

Methods. We compare our DTW similarity (c.f. Section 5.4) with the following algorithms for harmonic and time series similarity:

- **Tonal Pitch Step Distance (TPSD)** [103, 104], a state of the art method that measures the difference between the *Harmonic Profiles* (see \mathbf{q} in Section 5.4) of the given harmonic sequences. The difference is determined as the minimal area between the two time series, after considering all possible horizontal shifts. TPSD can handle sequences of different length, and has a time complexity of $\mathcal{O}(nm \log(n + m))$, where n and m denote the length of the compared chord sequences [8];
- **Longest Common Subsequence (LCS)** [397], a method expressing time series similarity based on their longest common subsequence. Similarity is calculated as the relative length of the longest common subsequence compared to the length of the shortest time series, thus ranging in $[0, 1]$. Using dynamic programming, LCS is bounded in $\mathcal{O}(n^2)$;
- **Soft Dynamic Time Warping (Soft DTW)** [80], a variant of DTW

Chapter 5. Uncovering Harmonic Similarity: From Musicological to Creative Exploration

Algorithm	TPS Mode	Stretch	Constraint	Normalise	Schubert		CASD		Schubert+CASD	
					First Tier	Second Tier	First Tier	Second Tier	First Tier	Second Tier
TPSD	offset	-	-	-	0.49	0.63	0.62	0.68	0.58	0.67
TPSD	profile	-	-	-	0.53	0.74	0.76	0.83	0.69	0.8
DTW	offset	stretch	-	-	0.94	0.98	0.53	0.67	0.66	0.76
DTW	profile	stretch	-	-	0.97	0.99	0.6	0.69	0.71	0.78
DTW	offset	stretch	sakoe_chiba	-	0.96	0.99	0.62	0.7	0.72	0.79
DTW	profile	stretch	sakoe_chiba	-	0.97	0.99	0.69	0.77	0.77	0.84
DTW	offset	stretch	itakura	-	0.96	0.99	0.55	0.65	0.68	0.75
DTW	profile	stretch	sakoe_chiba	yes	0.97	0.99	0.7	0.76	0.79	0.83
LCSS	offset	-	sakoe_chiba	-	0.38	0.61	0.03	0.07	0.14	0.24
LCSS	offset	-	itakura	-	0.7	0.8	0.14	0.23	0.31	0.41
SoftDTW	offset	stretch	-	-	0.93	0.97	0.55	0.69	0.67	0.77
SoftDTW	profile	stretch	sakoe_chiba	-	0.98	0.99	0.62	0.73	0.73	0.81

Table 5.2: Performance of similarity algorithms on cover song detection. The highlighted lines denote the best performing algorithms, while results in bold indicate the best performance obtained for the First Tier and Second Tier, respectively.

that allows for non-binary (fuzzy) alignments between time series, by using a soft-constraint. Soft DTW can be computed with quadratic time/space complexity.

All experiments are performed on the Harmonic Profile, in addition to an alternative formulation of the TPS time series, called *offsets*, where $q_i = \mathbf{tps}(x_i, x_{i-1})$ (chord offset distance).

For *DTW*, *LCSS* and *Soft DTW*, two types of constraints were also tested: *Sakoe-Chiba* and *Itakura*. Analogously to Sakoe-Chiba, the Itakura constraint sets a maximum distance for each point in the time series, making the algorithm more efficient, and reducing the risk of being trapped in local minima. Several parameter settings for the Sakoe-Chiba radius and Itakura band were tested, and the best results were obtained by setting them to 5 and 4, respectively. This parametrisation turned out to be optimal across all our experiments.

Each method was tested on sequences of original length (*no-stretch*) and after resampling to the shortest sequence. We also experimented with normalised time series (Section 5.4).

Dataset. We use two subsets of ChoCo (see Chapter 4) containing cover tracks: *Schubert Winterreise* [402] and *Chordify Annotator Subjectivity Dataset (CASD)* [227]. The former provides harmonic annotations for each of the 9 different performances of the same musical piece by Schubert. Similarly, CASD contains four annotations of the same performance, contributed by four different annotators. Chords from *Isophonics Dataset* [258] and *Jazz Audio-Aligned Harmony (JAAH)* [129] are also added to the reference dataset in order to add heterogeneity (different genres) and increase the complexity of the task.

5.4. Harmony: The Harmonic Memory

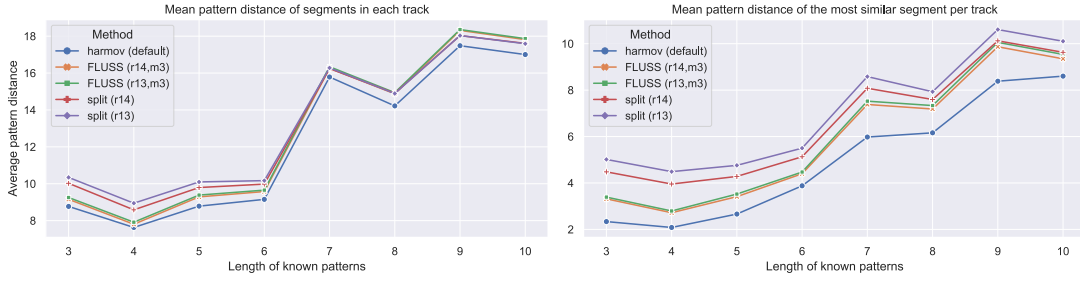


Figure 5.8: *Structural coverage of known patterns for each segmentation method, using Equation 5.11 (left), and Equation 5.12 (right). Results are reported as distances averaged per pattern group (a group contains known harmonic patterns of the same length).*

Results. Table 5.2 shows the results of this comparison and highlights the best performing algorithms. Results are presented for Schubert and CASD separately, and also in a third merged setup (Schubert+CASD). Notably, the performance of the DTW algorithm is significantly better for Schubert (one piece, multiple performances), while for CASD (one performance, multiple annotations), TPSD performs slightly better. The best results for the third setup are obtained using the Sakoe-Chiba DTW, using normalisation and resampling on the shortest sequence. It is also worth remarking that our implementation, besides being the most accurate overall, is also the most efficient approach, due to its linear complexity.

Structural coverage of known patterns

To validate our harmonic segmentation (Section 5.4), we measure the overlap between the resulting structures with a collection of well-known chordal patterns. This exemplifies the hypothesis that a good segmentation would maximise the “reuse” of harmonic patterns – as building blocks that can be found in other pieces.

Given a segmentation $\bar{\mathbf{q}} = \bar{\mathbf{q}}^1, \dots, \bar{\mathbf{q}}^P$ of a piece, with each $\bar{\mathbf{q}}^i$ containing a TPS time series, the overlap of $\bar{\mathbf{q}}$ with a dataset of known harmonic patterns \mathcal{P} is computed as:

$$o_M(\bar{\mathbf{q}}) = \frac{1}{T} \sum_{i=1}^T \min_{\mathbf{p} \in \mathcal{P}} d_{\text{DTW}}(\bar{\mathbf{q}}^i, \mathbf{p}), \quad (5.13)$$

$$o_B(\bar{\mathbf{q}}) = \min_{\bar{\mathbf{q}}^i \in \bar{\mathbf{q}}} \min_{\mathbf{p} \in \mathcal{P}} d_{\text{DTW}}(\bar{\mathbf{q}}^i, \mathbf{p}), \quad (5.14)$$

which differ in the aggregation function. The former measures the average pattern distance contributed by each structure in the segmentation; the latter, instead, only retains the distance of the most similar pattern that was matched to one of the structures. When $o_M = 0$, all segments are fully matched/found in \mathcal{P} ; whereas o_M is minimal when at least a segment matches a pattern in \mathcal{P} .

Methods. We compare our method (denoted as *harmov*) to fast low-cost unipotent semantic segmentation (FLUSS) [153] – a state of the art algorithm for time series segmentation defined on the Matrix Profile [416]. FLUSS annotates the time series with information about the likelihood of a regime change (a segment boundary); and is parameterised by a fixed window size m , and the number of segments to detect r . We also include a baseline splitting a time series in r uniform segments. Both methods operate on the TPS Profile of \mathbf{h} , and are optimised via grid search.

Datasets. We compute and evaluate the harmonic segmentations on a dataset comprising 320 chord progressions, obtained from randomly sampling 40 pieces per audio partition in ChoCo (see Chapter 4) (*isophonics*, *billboard*, *casd*, *schubert-winterreise*, *rcw-pop*, *uspop-2002*, *jaah*, *robbie-williams*). This yields a diversified (several genres, durations, etc.) yet representative sample of Harmory ($\approx 2\%$ of ChoCo); which prevents larger partitions from biasing the overall results. For \mathcal{P} , we assembled a dataset of known harmonic patterns from Impro-Visor [321], which is available on GitHub⁷. After filtration of trivial occurrences (e.g. chord uni-grams, sequences with repeated chord occurrences, etc.), the dataset counts 300 unique patterns spanning from 3 to 10 chord occurrences per pattern (the length of a chordal pattern).

Results. The structural coverage, computed for each segmentation method and aggregated for all known harmonic patterns of the same length, is reported in Figure 5.8. For both measures o_M , o_B , the segmentations produced by our method (*harmov*) produce the lowest distances – meaning that they show the highest overlap with the known harmonic patterns in \mathcal{P} . This behaviour is preserved for all pattern groups (the x-axis), and the gap with the other methods increases with pattern’s length. The second performing method is FLUSS, using $r = 14$ split regions and a window size of $m = 3$. However, for longer patterns, the latter performs comparably with a fixed sequence split (the other baseline). Finally, it is

⁷<https://github.com/Impro-Visor/Impro-Visor>

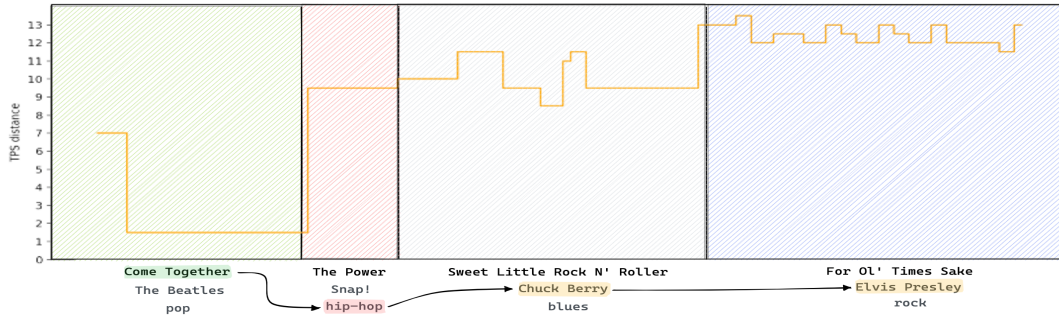


Figure 5.9: *Example of a generated chord progression using a pattern-based prompt. Given a first segment, each segment is chosen according to similarity to the subsequent one in the original sequence, and filtered according to arbitrary criteria. The second segment is taken from a song that has “hip-hop” as genre, while the next two segments are chosen by artist.*

worth remarking that results for all baselines are first optimised on a grid search; whereas we use the default parameterisation for *harmov*.

5.4.3 Avenues for machine creativity

We envisage various applications of Harmory across different tasks and use cases, ranging from music information retrieval and computational musicology, to creativity support for artistic workflows. The latter is the main focus of this work. However, we do not aim at improving the state of the art in music generation, but rather to provide a transparent system to support creative workflows [57].

Here, we show examples of trustworthy applications for pattern discovery, human-machine chord generation, and harmonic similarity. The latter is more of musicological interest, whereas the former are both creative use cases. Each application is demonstrated through a set of *Prompts*, expressed in natural language, which correspond to SPARQL query templates to interrogate the KG (Section 5.4). The latter are fully available on our repository⁸.

Pattern discovery

The traversal of the Harmonic Memory makes it possible to obtain granular information of the harmonic structure of songs. In particular, it is possible to explore the harmonic segments of each song, the patterns related to each segment, and the similarities with other patterns/segments found in other pieces.

⁸SPARQL queries: <https://github.com/smashub/harmory/tree/main/queries>

Chapter 5. Uncovering Harmonic Similarity: From Musicological to Creative Exploration

A composer may start with a harmonic pattern mind, and initiate a creative exploration of the KG by leveraging music and metadata.

Prompt 1 *For a given pattern, which are the tracks (titles, artists and genres) in which the pattern can be found?*

Prompt 2 *Given a music genre, what are the most frequent patterns?*

To support creative exploration, more complex prompts can be formulated in order to narrow down the search, and eventually discover surprising or unexpected outputs, if present.

Prompt 3 *Which harmonic patterns are used in “Michelle” by The Beatles, but also in a classical composition?*

Prompt 4 *Which patterns used by The Beatles in “Michelle” but not in “Hey Jude” contain at least a B flat major seventh chord?*

In the Harmory KG, we have included known patterns (as described in Section 5.4.2), which are labelled in such a way as to indicate their origin, mood, or harmonic function within the progression. These labelled harmonic fragments can be used as input for a query, e.g. for searching songs that contain them:

Prompt 5 *Which tracks include a dominant cycle in seven steps?*

Human-machine chord generation

Harmory also enables combinational creativity use cases. New progressions are generated by moving across patterns through temporal and similarity links, based on the given creative requirements. At generation time, this has the advantage of giving recognition to all artists that contributed to the new creation, as shown in Figure 5.9.

First, it can provide statistical information regarding variations of a given harmonic sequence. As these variations come from real pieces, it is also possible to leverage metadata for controlling the generation. To do this, a prompt can be formulated from a given (possibly new) harmonic sequence (or a part of it), to retrieve all the all harmonic sequences using the same pattern.

Prompt 6 *Given a chord sequence, which are its variations, and which tracks these variations belong to?*

Similarly, it is also possible to query the most similar (or most distant) harmonic sequences to a given one:

Prompt 7 *Given a chord sequence, which are its most similar chord sequences, sorted by similarity?*

These simple constructs already allow to generate new harmonic sequences, starting from either a known harmonic idea/pattern, or a full progression. If starting from a full progression, one way is to identify the first harmonic segment that makes it up. From this point, transitions can be made using *similarity relations*, while taking into account the order of the different segments (*temporal connections*) and their tonality. For example, starting from the first harmonic segment of a song (a priming sequence), one can then generate a continuation by identifying similar sequences to the next sequence, filtering them by tonality (or/and by artist, genre, title) and repeat this process recursively for a number of steps, criteria, or with the supervision/control of the user.

Prompt 8 *Create a progression starting with “Michelle” by The Beatles, continuing with a segment found in a classical piece of music, and then continuing with another by Chet Baker.*

Harmonic similarity

From a musicological perspective, the KG can also be used to analyse similarity relations between tracks – by leveraging the local information relating harmonic structures. This also allows for the formal definition of similarity functions (depending on a genre- or task- specific notions) by using logical operators (SPARQL syntax) over harmonic segments/patterns. An example is given below.

Prompt 9 *Given a track, which tracks contain patterns with a distance of less than 0.2, each having the same order?*

As expected, the results of this query are almost exclusively cover songs of the given track. Nevertheless, a similarity function can be defined to be less strict, and hence more explorative. For instance, the similarity function below uses a higher similarity threshold for patterns, and does not constrain on the order of segments.

Prompt 10 *Given a track, which tracks contain patterns with a distance of less than 0.5, regardless of their order?*

5.5 Conclusion

The creation of large, harmonized corpora of symbolic harmonic annotations opens up unprecedented opportunities for exploring harmonic data at scale, as discussed in Section 4.5. Integrating diverse datasets enables analyses across a broad spectrum of musical genres, styles, and artist collections, providing insights into harmonic content that were previously limited by fragmented data sources.

To fully leverage these harmonized corpora, a structured approach to similarity is essential for analyzing and interpreting harmonic content across extensive datasets. Unlike traditional methods that rely on general metadata or surface-level features, content-based harmonic similarity offers a deeper layer of analysis by identifying intrinsic musical patterns and relationships. This approach enables the detection of nuanced harmonic structures and recurring patterns that define genres, stylistic shifts, and even the idiosyncrasies of particular artists or historical periods.

In this chapter, we investigate how the large-scale, harmonized corpora introduced in Chapters 3 and 4, alongside novel similarity measures, can support both musicological research and creative generation. To this end, we propose and evaluate two key contributions: *LHARP* and *Harmory*.

LHARP is a method developed to capture local harmonic similarities across pieces. Unlike traditional global similarity measures, LHARP focuses on detecting recurring harmonic patterns within musical sequences, allowing researchers to explore harmonic nuances that are often genre- or artist-specific. This approach offers a more flexible and interpretable similarity measure, particularly suited for musicological tasks that require detailed comparisons of harmonic structures across different works (Section 5.3).

Harmory, or the Harmonic Memory, is a knowledge graph designed to support computational creativity by organizing harmonic patterns within a structured, musically meaningful space. Leveraging cognitive and musicological models, Harmory captures both temporal relationships and structural similarities between harmonic sequences, linking them within a broader harmonic landscape. By facilitating transparent and accountable access to diverse harmonic structures, Harmory provides a valuable resource for compositional assistance, allowing musicians to explore and experiment with harmonic ideas across different genres and styles (Section 5.4).

5.5.1 Limitations and Future Work

While LHARP demonstrates potential for generalizing to arbitrary symbolic sequences with structural similarities to music, several limitations remain, which define directions for future work. A primary concern lies in the data pre-processing and encoding steps. Specifically, transposing all pieces to a common key, though practical for similarity calculations, may raise objections among music experts who argue that it could alter the musical texture, compromising the integrity of the original (non-transposed) version. Additionally, the encoding method, which decomposes chords into individual pitches, may overlook context-specific nuances. For example, two distinct chord labels producing the same sound might have unique harmonic functions that are flattened by pitch-based encoding. These limitations represent simplifying assumptions, yet we are still actively gathering feedback to validate their applicability and explore possible refinements.

These challenges are partially addressed in Harmory’s similarity function, which leverages the TPS for encoding chords. While effective and musicologically grounded, this approach remains an incomplete and context-unaware representation of musical harmony. Future research could involve exploring advanced representation learning techniques to encode harmonic sequences more contextually and robustly [275, 113].

Moreover, as LHARP and Harmory currently provide single measures of similarity, their effectiveness is naturally limited to the tasks and use cases tested (e.g. cover song detection). Expanding the experimental framework to new tasks will be essential for assessing the broader applicability of these methods. Additionally, future work will involve developing novel similarity functions that can adapt to a wider range of musicological and creative applications, thereby enhancing the versatility and depth of both LHARP and Harmory in supporting diverse aspects of music analysis.

CHAPTER 6

Exploring Symbolic Limitations: Multimodal Strategies for Enhanced Harmonic Analysis

6.1 Introduction

In the previous chapters, we discussed how the large corpora of musical data proposed in Chapters 3 and 4 can be leveraged in standard MIR tasks. Specifically, in Chapter 5, we explored the task of harmonic similarity and its applications, demonstrating the potential of these corpora to tackle new challenges in MIR.

However, despite the strides made, the data integration process finalised at the creation of large datasets also exposes several limitations. While large music corpora can serve as powerful tools for advancing harmonic analysis and other MIR tasks, they incorporate important gaps that still need to be addressed to unlock the full potential of data-driven approaches in music research.

These limitations manifest particularly in two critical areas: the inherent imbalance in chord distributions, and the subjective nature of harmonic annotations. As we will explore in this chapter, these challenges impact the development of robust MIR systems, especially in tasks like ACE and harmonic analysis.

6.1.1 Limits and Challenges to Harmonic Data Integration

Recently, a comprehensive survey [302] discussed the advancements and challenges encountered throughout twenty years of research in harmonic analysis. Among the array of challenges identified, two emerge as particularly significant, both intimately tied to the intricate nature of harmonic content and its representation within audio signals: (i) the scarce diversity and balancing of the available datasets and (ii) the inherent ambiguity and subjectivity of chord annotations.

In this chapter, we will explore these challenges in depth, as they present critical barriers to the further advancement of MIR systems.

Diversity and Balancing of Available Datasets

One of the most significant challenges in current MIR systems is the lack of diversity in the datasets available for tasks such as harmonic similarity and chord analysis. The Chord Corpus (ChoCo), introduced in Chapter 4, represents the largest and most comprehensive dataset of its kind, integrating over 20,000 harmonic annotations. It includes annotations from a four diverse main musical styles, such as pop, rock, jazz, and classical, making it one of the most diverse datasets in the field. However, despite its breadth, the majority of the data skews heavily toward mainstream Western genres, leaving significant gaps in the representation of other musical traditions and styles.

For instance, the majority of the dataset – nearly 80%—comes from pop and rock music, while jazz and classical account for a much smaller portion. This lack of diversity has a direct impact over the distribution of chords within the dataset. In Western popular music, certain chord types—such as major, minor, dominant seventh, and major seventh – are disproportionately frequent. Meanwhile, more complex or extended chord types, which are prominent for example in jazz music, are under-represented. This imbalance in chord distribution, often referred to as *chord vocabulary imbalance* in literature [227], limits the model’s ability to correctly handle infrequent chords, posing a strong bias towards the most represented ones.

For instance, in *ChoCo* approximately 74.9% of the distribution of the 8064 distinct chord classes is dominated by just 5 chord types, as illustrated in Figure 6.1. These limitations are especially relevant when this data is used for ML and DL applications, where algorithms notoriously struggle dealing with unbalanced distributions [352].

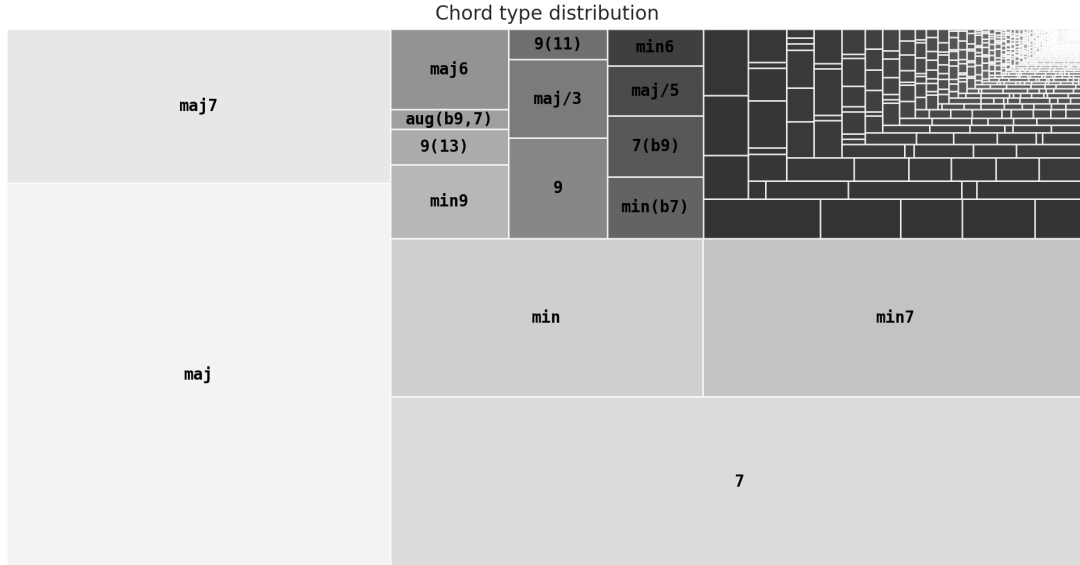


Figure 6.1: *Distribution of chord types in the ChoCo dataset.*

Ambiguity and Subjectivity in Chord Annotations

Another critical challenge is inter-annotator agreement, which arises from the inherent ambiguity in what constitutes a chord from a musical perspective and the subjective nature of human annotation processes. For example, a clear distinction between a chord sequence and a melodic line can be subject to individual interpretation. Moreover, there is significant variance among annotators regarding the level of detail in annotating chord sequences [227].

The issue of inter-annotator agreement arises from the inherent ambiguity in what constitutes a chord from a musical perspective and the subjective nature of human annotation processes. For example, a clear distinction between a chord sequence and a melodic line can be subject to individual interpretation. Moreover, there is significant variance among annotators regarding the level of detail in annotating chord sequences, such as the inclusion of rapid approach chords or arpeggiated chords. Furthermore, annotators might be biased towards the melodic and harmonic line played by a specific instrument, such as piano or guitar, rather than considering the broader harmonic profile of the piece [227].

One particularly famous example of subjectivity in chord annotation is the opening chord of The Beatles’ “A Hard Day’s Night” [227]. This chord has generated decades of debate among music theorists and musicians, with no single definitive answer, bringing Beatles experts like Pedler [303] to observe that this chord has taken on a “holy grail” status in popular music analysis. George Har-

arrison once described it as an “F with a G on top,” played on his 12-string guitar, but left the interpretation of the bass note to Paul McCartney. Gary Moore suggested it was a **G7sus4**, to which Harrison disagreed and demonstrated a different fingering. Other music theorists have offered varying analyses, including **G11sus4**, **Gsus4/D**, and even D minor 11th.

Despite these ambiguities, single reference annotations are often used as the de facto ground truth in computational studies of harmony [227]. These annotations are treated as objective labels, against which algorithmic outputs are evaluated. Yet, relying on a single reference ignores the inherent variability in human perception and the possibility of valid alternative interpretations.

Various studies have investigated inter-annotator agreement in chord annotation, aiming to measure the magnitude of the agreement in datasets specifically created for this purpose. Clercq et al. [97] observe an inter-annotator agreement rate of 94% for the root note between two different annotations of the top 20 tracks from Rolling Stone magazine’s list of the *500 Greatest Songs of All Time*. In contrast, Koops et al. [227] report an inter-annotator agreement rate of 76% for the root note on four different annotations of a 50-song subset of the Billboard dataset [48], while the agreement for more complex chords lowers to 52%.

6.1.2 The Need for Multimodality

The challenges discussed above, particularly the limitations of symbolic chord annotations and the issue of inter-annotator agreement, highlight a critical need for a more robust and flexible approach to analysing harmonic data. In this thesis, we propose a multimodal approach that integrates both symbolic harmonic annotations and audio signals to overcome these limitations.

Multimodal integration of audio and symbolic data offers several advantages that make it a powerful solution for addressing the problems inherent in chord annotation:

1. A multimodal approach enables us to tackle tasks that inherently require audio data, such as Audio Chord Estimation (ACE);
2. Multimodal systems replicate the process human annotators follow when transcribing chords. Human annotators typically rely on both the audio signal and their musical knowledge to determine harmonic structures. By incorporating audio data, multimodal systems can more closely reflect this

human approach, improving performance in tasks such as resolving inter-annotator disagreement by considering the auditory context in which chords are perceived;

3. Audio complements symbolic annotations by providing information that is difficult or impossible to capture symbolically. For example, features like timbre, dynamics, and articulation are essential aspects of music that contribute significantly to its harmonic character, yet they are often left out of purely symbolic representations [396];
4. This approach is aligned with growing advocacy within the MIR community for integrating symbolic data into audio analysis. As symbolic and subsymbolic methods each offer distinct advantages, balancing these approaches in multimodal systems provides a more holistic view of musical content [66].

Audio-to-Chord Alignment

To create a multimodal dataset that integrates both symbolic chord annotations and audio signals, the first critical step is retrieving the corresponding audio for the existing chord annotations. Once the audio is obtained, it is necessary to align audio and symbolic representations.

However, the availability of audio-aligned annotations within current corpora is highly limited. In the case of ChoCo, less than 12% of the 20,000 annotated tracks are aligned with audio. This highlights the need for an efficient method to align audio with chord annotations.

An effective audio-to-chord alignment approach would not only bridge the gap for existing annotations but also enable the generation of new multimodal data. One promising avenue is to leverage the vast repositories of crowd-sourced chord annotations available online. Platforms such as Ultimate Guitar¹, e-chords², and Chordie³ collectively house millions of annotated songs, offering a wide variety of genres that are currently underrepresented in existing MIR datasets. These genres include electronic, metal, hip hop, reggae, and country, among others, providing an opportunity to broaden the scope of the dataset beyond the current mainstream genres.

Moreover, these repositories often contain multiple versions of the same song,

¹<https://www.ultimate-guitar.com/>

²<https://www.e-chords.com/>

³<https://www.chordie.com/>

each with slightly different interpretations. This abundance of versions could be invaluable for analysing the subjectivity in chord annotations, as suggested in works like [226, 227]. However, a significant limitation is that these crowd-sourced chord annotations typically lack any timing or duration information, offering only chord lists and sometimes lyrics. This deficiency severely restricts their use for MIR-related tasks that rely on temporal alignment between audio and symbolic data.

These challenges underscore the need for a system capable of aligning chord annotations with audio recordings. Currently, no model has been explicitly developed for this purpose. While existing audio-to-score alignment techniques, such as those based on MIR algorithms [288], usually provide good quality alignments, they typically require some preliminary weak alignment between the score and audio, which is often not available in the case of chord annotations, especially when dealing with crowd-sourced data. These methods struggle to handle chord annotations with no temporal information, rendering them insufficient for the large-scale, automated alignment of chord lists with audio.

Audio Chord Estimation (ACE)

The proposed multimodal dataset, which integrates homogenised symbolic chord annotations from diverse sources and aligns them with audio signals, holds significant potential for advancing ACE – a critical task in MIR. ACE automates the transcription of chords directly from audio recordings, offering a scalable solution for music transcription and analysis. Its applications are far-reaching, impacting fields such as music analysis, musicology, content-based retrieval, and music education.

Over the past two decades, research in ACE has made considerable progress, leading to notable improvements in the accuracy and efficiency of chord transcription [302]. However, despite these advancements, recent performance gains have stagnated, prompting some researchers to suggest that the task has reached a “glass ceiling” [56]. Notably, increasing the amount of training data and scaling computational resources have not resulted in significant improvements in ACE performance [302]. This plateau is largely due to ongoing challenges such as the aforementioned chord vocabulary imbalance and inter-annotator disagreement.

This thesis seeks to explore strategies for overcoming these obstacles, with a focus on improving our understanding of inter-annotator agreement, enhancing

ACE performance, and better capturing harmonic representations from audio [23].

We begin by evaluating existing metrics for inter-annotator agreement in chord annotations [97, 227]. Current evaluations typically rely on binary metrics, where a match between two labels is scored as one and any mismatch is penalised with a score of zero. However, as noted by [266], treating all discrepancies with equal severity can result in unfair assessments. Binary evaluations often fail to account for shared harmonic features between chords that, while annotated differently, exhibit meaningful similarities. For example, a mismatch between a *G7* and a *Gsus4* may be treated as a complete error, despite both chords sharing significant harmonic tension and resolution characteristics.

To address these issues, we propose leveraging music theory to enhance the model’s understanding of chord annotations. By embedding theoretical concepts such as consonance and dissonance, we aim to provide more context-aware interpretations of chord sequences. This theoretical framework will allow the model to better distinguish between similar chords and offer more nuanced interpretations of ambiguous or subjective harmonic structures.

Finally, we will evaluate the effectiveness of this approach in improving ACE and the resulting harmonic representation from the audio signal. By combining both symbolic and audio data, along with a deeper integration of music theory, we aim to overcome current limitations and establish a more robust framework for automatic chord transcription.

6.1.3 Our Contribution

In this chapter, we introduce two significant contributions aimed at addressing the limitations in chord annotation alignment and ACE through multimodal approaches, in response to **RQ4** (Section 1.1.4) and **RQ5** (Section 1.1.5) respectively. These contributions were published in [316] and are further expanded in a paper currently under review [317].

First, we present *ChordSync*, a novel method for aligning chord annotations to audio tracks without the need for preliminary weak alignment. By leveraging the power of the conformer architecture [170], ChordSync enables the seamless synchronization of chord annotations with audio signals from large-scale datasets. This approach opens up the possibility of expanding audio-aligned chord datasets using existing resources that lack timing and duration information, such as those found on crowd-sourced platforms like UltimateGuitar. We provide a pre-trained

model and a user-friendly library to help users effortlessly align chord annotations with audio recordings. This advancement is particularly valuable for MIR tasks that rely on multimodal data, such as music structure analysis, and offers enhanced opportunities for music learning experiences by aligning harmonic content with auditory cues.

Additionally, we contribute a comprehensive analysis of inter-annotator agreement in chord annotations using non-binary metrics, building upon the work of McLeod et al. [266]. Traditional binary metrics often penalize disagreements too harshly, without accounting for harmonic similarities between annotations. Our analysis demonstrates that incorporating distance metrics based on perceptual consonance significantly improves agreement scores. This insight provides a deeper understanding of how subjective harmonic interpretations can be better captured and evaluated in MIR systems.

Building upon these findings, we propose a novel approach to ACE that integrates consonance-based label smoothing [287] and focal loss mechanisms [246]. To address the persistent issue of chord vocabulary imbalance, we introduce a method inspired by McFee et al. [263], where chord root, bass, and note activations are classified separately, allowing the final predicted chord label to be derived from decoding these components rather than imposing a fixed vocabulary. This approach provides greater flexibility and adaptability, enabling the model to capture a wider range of harmonic structures beyond common chord types.

Our proposed ACE model is built upon the conformer architecture[170], that has been recently adopted for different audio-based tasks [378, 410]. By leveraging this architecture, we demonstrate that our model learns more effective representations of harmonic content and outperforms state-of-the-art ACE methods when evaluated using consonance-based distance metrics.

In summary, our contributions in this chapter are twofold: we introduce a robust solution for chord-to-audio alignment with ChordSync, and we propose an enhanced method for Chord Estimation that addresses key challenges such as chord vocabulary imbalance and inter-annotator agreement through a multimodal, theory-informed approach.

6.1.4 Chapter Structure

This chapter is structured as follows. In Section 6.2, we review the state of the art for both audio-to-score alignment techniques and Audio Chord Estimation.

Section 6.3 introduces our novel alignment method, ChordSync, and provides examples of how it can be used to integrate the ChoCo dataset with crowd-sourced chord annotations. Next, in Section 6.4, we analyse inter-annotator agreement using different non-binary metrics, emphasizing the role of consonance and dissonance in improving agreement analysis. Section 6.5 presents our proposed ACE approach, including the use of consonance-based label smoothing and focal loss mechanisms to address challenges such as chord vocabulary imbalance. Finally, Section 6.6 concludes the chapter, summarizing the main contributions and insights.

6.2 Related Work

6.2.1 Alignment Techniques

Audio-to-Score Alignment

The task of aligning audio to symbolic music, commonly known as *audio-to-score alignment* (A2SA), has been primarily addressed by *Dynamic Time Warping* (DTW) algorithms [279], as they are particularly effective for sequence alignment tasks. Thus, various DTW-based alignment methods have been proposed to align audio with different symbolic music formats, such as MIDI [322], often integrating additional techniques and diverse signal representations to improve alignment accuracy [51, 338].

A differentiable variant of DTW, *SoftDTW*, has been recently used as the loss function within neural network architectures, mainly for multi-pitch estimation tasks [232, 423]. However, a general limitation of the DTW-based approaches is their reliance on weak-aligned data to perform the alignment. This requirement renders them unsuitable for contexts without prior alignment information.

Other deep-learning methods have been investigated for audio-to-score alignment, including leveraging automatic transcription techniques [360] and training audio features tailored explicitly for alignment tasks [212].

The only previously proposed approach for aligning audio with chord annotations uses Hidden Markov Models (HMM) and is part of an ACR workflow [411]. Also related to our work is the *Harmonic Change Detector* (HCD), introduced in [180] and subsequently revisited and improved in [108, 329], for detecting harmonic changes within the audio signal, including chord changes. However, the number of harmonic changes within the audio signal often exceeds the number of chord

changes, posing challenges for using these algorithms directly for audio-to-chord alignment.

Lyrics-to-Audio Alignment

Another form of alignment pertinent to our work is the audio-to-lyrics alignment task, which seeks to determine the corresponding locations in a song recording of its lyrics at various levels such as line, word, or phoneme [354]. Existing methods for this task are commonly adapted from automatic speech recognition (ASR) [172, 368], despite the inherent complexity of singing voices compared to speech [200], and typically make use of acoustic models trained to recognise the phonetic content of the audio signal at various levels of granularity. Some recent works have adopted the Connectionist Temporal Classification (CTC) loss [164], training the acoustic model in an end-to-end fashion [368].

6.2.2 Audio Chord Estimation (ACE)

Since the seminal work by Fujishima from 1999 [146], most chord recognition systems applied a knowledge-driven approach [267], involving the extraction of acoustic features, such as chroma [259] or Tonnentz [202], followed by classification or template matching techniques, such as HMMs [21], Dynamic Bayesian Networks (DBNs) [259], or Conditional Random Fields (CRFs) [228].

With the emergence of deep learning, various architectures have been explored for the task, including CNNs [263, 228], Recurrent Neural Networks (RNNs) [357], Convolutional Recurrent Neural Networks (CRNNs) [211], and Transformers [301]. While deep-learning approaches have surpassed traditional knowledge-driven ones, several challenges must be tackled.

Most of the proposed approaches to addressing the chord class imbalance challenge can be divided into two categories: chord simplification and chord decomposition. The former reduces the size of the chord vocabulary by converting complex chord labels into simpler representations. Notably, the vast majority of studies have adopted restricted vocabularies of approximately 25 symbols, encompassing major-minor chords along with placeholders for other chord types X and silence N [146, 267]. Chord decomposition strategies focus on predicting the chord constituting components separately, typically the bass, root, and note activations (often also 7th, 9th, 11th, and 13th), and then map them to templates to predict the final chord [263, 412, 211]. Some additional approaches do not fall into these

two categories, like addressing the unequal distribution of chords through a balanced learning process [110], or using a curriculum learning training scheme to begin with simple chord qualities and then move to more complex and less common ones [343]. The inter-annotator agreement in chord annotation continues to pose a significant challenge. Despite existing diagnoses and quantification of this phenomenon in the literature [227, 97], definitive solutions have yet to emerge. Clercq et al. [97] observe an inter-annotator agreement rate of 94% for the root note between two different annotations of the top 20 tracks from Rolling Stone magazine’s list of the *500 Greatest Songs of All Time*. In contrast, Koops et al. [227] report an inter-annotator agreement rate of 76% for the root note on four different annotations of a 50-song subset of the Billboard dataset [48]. To address this challenge, Koops et al. [226, 227] propose a novel approach based on chord label personalisation. Instead of employing a uniform set of chord labels for all users of an ACE system, this method advocates for the customisation of chord labels to suit individual preferences and vocabulary. The process starts with the calculation of Shared Harmonic Interval Profiles (SHIP) representations derived from multiple chord label reference annotations corresponding to the Constant-Q Transform (CQT) frames. Then, a deep neural network is trained to learn these features from audio. Finally, personalised chord labels are generated, tailored to each annotator’s specific vocabulary and preferences.

While this approach offers valuable insights, it requires multiple annotations of individual tracks, which restricts its applicability due to the scarcity of such datasets [225]. In contrast, the method proposed in this thesis endeavours to develop generalised harmonic representations by leveraging principles derived from music theory, thus circumventing the need for datasets with multiple annotations. Our method applies Label Smoothing (LS), a technique employed to enhance the generalisation and learning speed of multi-class neural networks. Originally proposed in [376], LS redistributes a portion of the probability mass from the observed class to other classes, thereby softening the distribution and generating what is referred to as *soft targets*. This regularisation method has found widespread application in various state-of-the-art models across domains such as image classification, language translation, and speech recognition. LS has also been tested for music classification tasks [46], proving to improve performance and reduce overfitting in small network training. While LS primarily serves as a regularisation technique, numerous studies have delved into its potential for

encoding meaningful relationships among different categories. For instance, in [250], authors propose an impactful method for generating more reliable soft labels that explicitly consider the relationships among various categories. Similarly, in [245], a novel approach known as *label relaxation* is introduced, which involves replacing a degenerate probability distribution associated with an observed class label, not by a single smoothed distribution but rather by a larger set of candidate distributions.

6.2.3 Conformer-based Approaches

The conformer architecture [170] has recently emerged in Automatic Speech Recognition (ASR) as a novel architecture to effectively model global and local audio dependencies by leveraging a combination of CNNs and Transformer architectures. It has showcased remarkable success across various tasks not only in speech [63] but also in music [410], including melodic transcription [378], representation learning [127], and music audio enhancement [58].

6.3 ChordSync: Conformer-Based Alignment of Chord Annotations to Music Audio

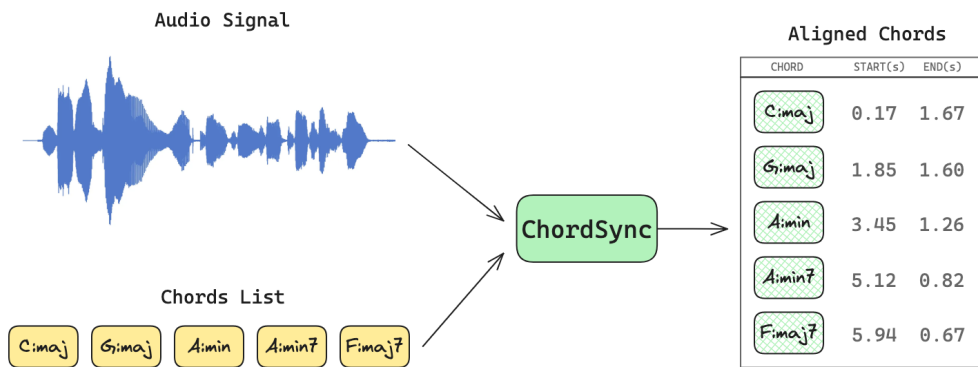


Figure 6.2: Basic schema of *ChordSync*: The model processes a list of chords alongside the audio signal, producing time-aligned chords as output.

This section describes *ChordSync*, our proposed conformer-based model for audio-to-chord alignment. It implements an acoustic model for estimating the frame-wise probabilities of chord labels, which are then fed to a forced-alignment decoder, along with the list of chord labels to align, as illustrated in Figure 6.2. Figure 6.3 illustrates the three primary steps implemented by the model: pre-

6.3. ChordSync: Conformer-Based Alignment of Chord Annotations to Music Audio

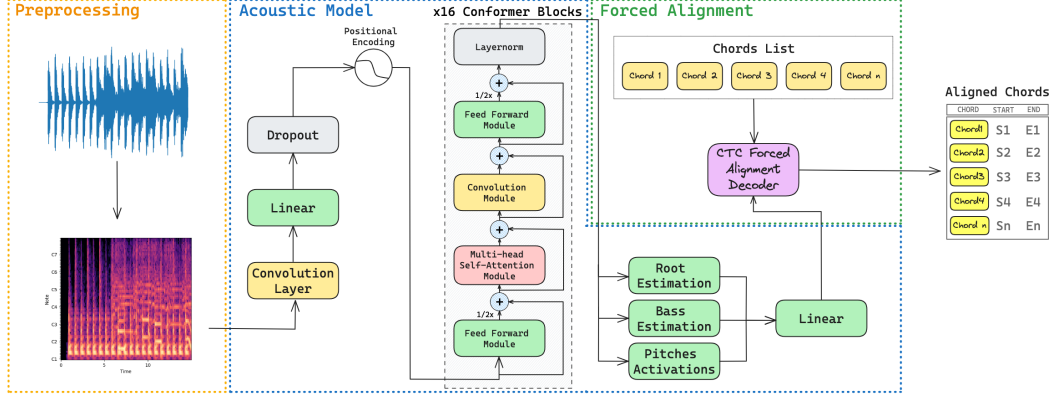


Figure 6.3: Architecture of ChordSync: (i) The audio signal undergoes preprocessing to Constant-Q Transform (yellow box); (ii) The preprocessed audio serves as input for training the conformer-based acoustic model (blue box); and (iii) The model output probabilities, along with the list of chord labels for alignment, is fed into a Connectionist Temporal Classification (CTC) forced alignment module (green box), which outputs the aligned chord labels.

processing and data augmentation (Section 6.3.2), the acoustic model used during training, and the forced alignment decoder (Section 6.3.3). The software implementation and a pre-trained model are available on a GitHub repository⁴.

6.3.1 Problem Statement

Let $X = \{x_1, \dots, x_N\}$ be a frame-level sequence of acoustic features extracted from the input audio, where $x_n \in \mathbb{R}^D$ represents a D-dimensional feature vector, and N indicates the total number of frames within the sequence. Let $C = \{c_1, \dots, c_M\}$ be the input list of chord labels encoded into integer values, where $c_m \in \mathbb{Z}^K$, K denotes the size of the chord vocabulary, and M is the length of the chord sequence. The list of chord labels is upsampled to match the length of the audio sequence N . This upsampling is performed uniformly, assuming each chord has a duration approximately equal to N/M . Specifically, each chord label c_m is repeated for approximately N/M frames to produce the sequence $Z = \{z_1, \dots, z_N\}$, where $z_m \in \mathbb{Z}^K$. Thus, we train an acoustic model to optimise the following equation:

$$Z^* = \underset{z}{\operatorname{argmax}} p(Z|X), \quad (6.1)$$

⁴<https://github.com/andreamust/ChordSync>

where Z^* represents the optimal sequence of chord labels that maximises the posterior probability $p(Z|X)$, given the input sequence X . Note that X and Z are aligned at the frame level, and $p(X|Z)$ is evaluated by estimating the frame-wise posterior probability $p(x_n|z_n)$.

The output probabilities $p(X|Z)$ from the acoustic model are then fed to a CTC forced alignment decoder, which estimates the best alignment between the sequence of acoustic features X and the list of chord labels C :

$$A^* = \operatorname{argmax}_a p(A|X, C), \quad (6.2)$$

where A^* represents the optimal alignment between X and C that maximises the posterior probability $p(A|X, C)$.

In this way, the decoder generates the aligned chord labels with respect to the audio signal.

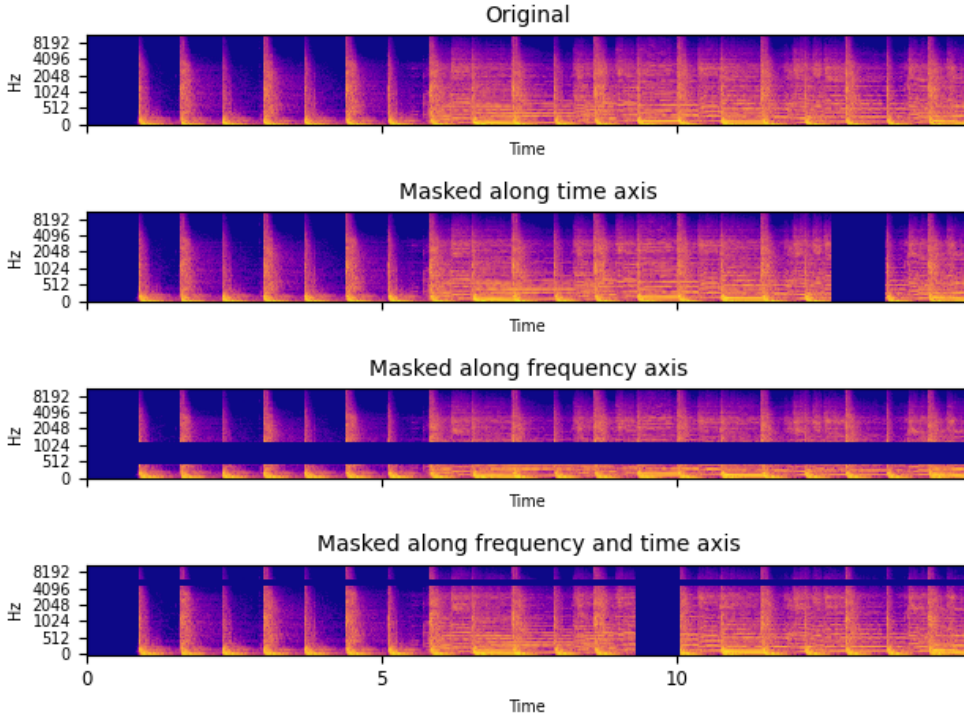


Figure 6.4: *Data augmentation policies used for ChordSync, all adapted from [300]. The CQT used as input for the model is randomly masked either: (i) along the time axis (second image); (ii) along the frequency axis (third image); or (iii) both time and frequency axis (fourth image).*

6.3.2 Preprocessing

For the input audio data, a standard pre-processing pipeline is implemented. The audio is first resampled to a sampling rate of 22050 Hz, and a hop size of 2048 is applied. Then, the Constant-Q Transform (CQT) features are calculated on 6 octaves starting from $C1$, with 24 bins per octave, resulting in a total of 144 bins.

The audio data used for training undergoes data augmentation by applying (i) time masking and (ii) frequency masking directly to the audio features, as proposed in *SpecAugment* for end-to-end ASR [300]. Frequency masking involves masking a random number of consecutive mel frequency channels along the frequency axis, whilst time masking involves masking a random number of frames along the time axis. Figure 6.4 shows examples of the implemented augmentation strategies.

During training, each audio excerpt in the training set undergoes augmentation, where either one of the transformations (frequency masking or time masking) or both are applied, and the choice of augmentation technique is determined randomly with equal probability.

Chord labels are numerically encoded into integer values and upsampled to match the length of the audio sequence N . The upsampling is performed using the `pumpp` library⁵. Figure 6.5 shows how chord labels are converted and sampled. The size of the chord vocabulary K results from the linear combination of the 12 pitches, representing the chromatic scale, with chord qualities such as {maj, min, 7, dim, dim7, hdim7, aug, min7, maj7, maj6, min6, minmaj7, sus2, sus4}, plus an additional chord symbol N representing silence or no chord.

6.3.3 Conformer-based Acoustic Model

The acoustic model we adopt is an adaptation of the original Conformer architecture [170], where the audio encoder processes the input through a convolutional module followed by a series of Conformer blocks.

The convolutional module comprises a convolution layer, a fully connected layer, and a dropout layer. The convolutional module serves as the initial feature extractor, capturing local patterns within the input CQT. Dropout regularisation is applied by randomly deactivating units during training to reduce overfitting. Additionally, we incorporate positional encoding, as proposed in the original trans-

⁵<https://github.com/bmcftee/pumpp>.

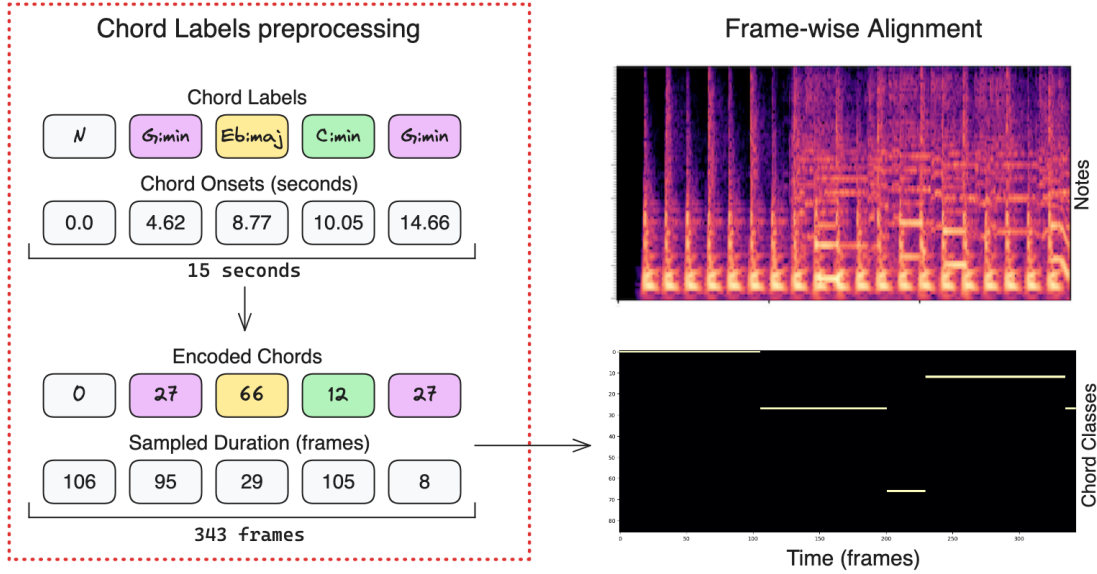


Figure 6.5: Workflow of the pre-processing applied to the chord labels. Chord labels are numerically encoded and upsampled to match the length of the CQT.

former architecture paper [393].

A Conformer block is composed of four modules stacked together: a feed-forward module, a self-attention module, a convolution module, and a second feed-forward module at the end. In the original Conformer paper, the authors explore three different sizes of the Conformer architecture: *S* (small), *M* (medium), and *L* (large), with different numbers of layers, hidden units, and other parameters. For our implementation, we opt for the *M* architecture, which comprises a 16 encoder layer with a dimension of 256, 4 attention heads, and a convolutional kernel size of 32. While the original paper observed significant improvements when transitioning from the *S* to the *M* variant, our experimentation yielded little improvements from *M* to *L*.

To handle the large dimensionality of the vocabulary, we use an architecture similar to that proposed by [263], in which root notes, bass notes, and all pitch activations of the chord are predicted. Subsequently, these probabilities are passed to a feed-forward layer, which converts these three probabilities into the likelihood of the chord with respect to the vocabulary K , similarly to what was proposed by [343].

For training, we employ cross-entropy loss and optimise using the AdamW optimiser. We utilise a cosine annealed warm restart learning scheduler to manage learning rates. Learning rate schedulers proved effective in training audio data,

6.3. ChordSync: Conformer-Based Alignment of Chord Annotations to Music Audio

especially with augmented data [300]. Finally, we applied early stopping by halting the training if the loss failed to decrease for over 20 epochs to prevent overfitting.

The output of the trained acoustic model is shown in Figure 6.6.

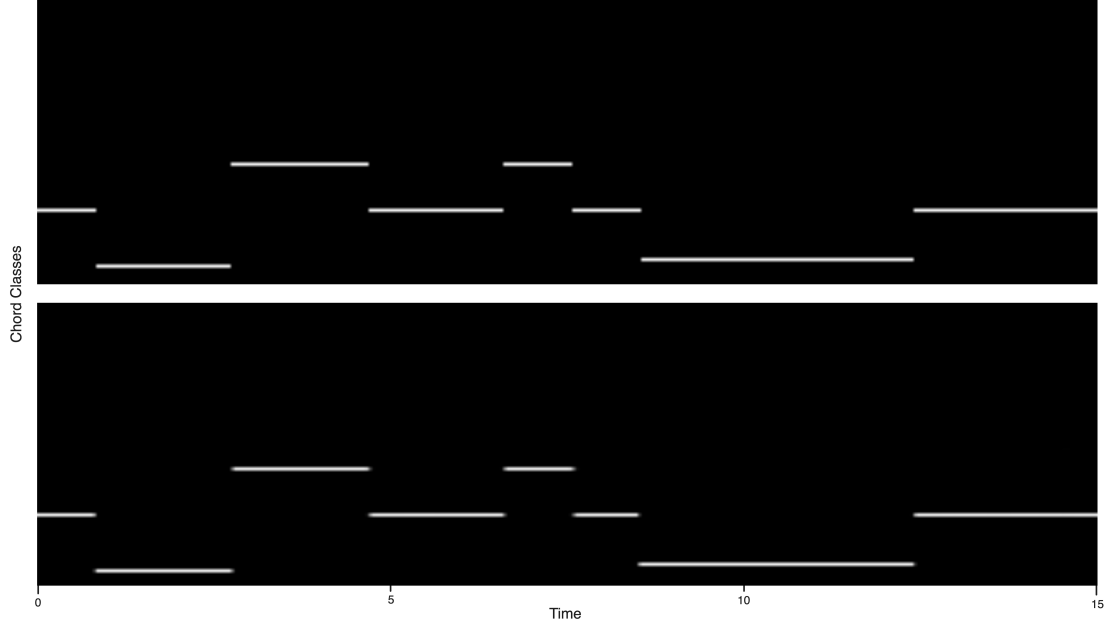


Figure 6.6: *Example of the chord classes predicted by the conformer-based acoustic model. The upper panel displays the target labels, while the lower panel showcases the probabilities predicted by the model.*

Forced Alignment

To estimate the best alignment between the acoustic features X and the chord labels C , we utilise the Connectionist Temporal Classification (CTC) objective function [164], which computes the probability of a given alignment between the input features and output labels. The CTC objective function is defined as follows:

$$p(C|X) = \sum_{A \in \mathcal{A}_{X,C}} p(a_t|X), \quad (6.3)$$

where $\mathcal{A}_{X,C}$ denotes the set of all possible alignments that produce the label sequence C , and $p(a_t|X)$ represents the probability of alignment a_t given the input sequence X .

The probability of alignment a_t given X is computed as the sum of probabilities of all paths a'_t that correspond to a_t after collapsing repeated labels and blank symbols:

$$p(a_t|X) = \prod_{t=1}^T p_t(\pi_t|X), \quad (6.4)$$

where T is the length of the alignment, and $p(\pi_t|X)$ is the probability of the t -th symbol in the alignment path π given the input sequence X .

6.3.4 Evaluation

Due to the lack of established methodologies to address the chord-to-score alignment task, conducting a comparative evaluation with existing state-of-the-art techniques presents some challenges. Therefore, to gauge the effectiveness of the proposed methodologies, we use alternative approaches performing analogous albeit slightly dissimilar methods for comparison and conduct two different experiments. The first aims to evaluate the model’s capability in detecting chord boundaries, while the second compares it to a traditional DTW-based alignment.

All experiments were carried out using a subset of ChoCo (c.f. Chapter 4), for which we select only partitions annotated on audio, i.e. expressing temporal information such as onsets and duration in seconds. Table 6.1 presents a summary of all ChoCo partitions employed for training and evaluation.

Audio files corresponding to each ChoCo annotation were obtained automatically from the available metadata in the original datasets. This was necessary as only a small portion of the datasets offer external links to the original audio sources used for chord annotation. Since the automatic retrieval process depends on sometimes sparse and incomplete metadata, the validity of the audio files was manually verified on randomly selected samples. The complete dataset consists

Dataset	Dataset	Genre	#Tracks	Reference
<i>Isophonics</i>		pop, rock	300	[258]
<i>Billboard</i>		pop	740	[48]
<i>Chordify</i>		pop	50	[225]
<i>Robbie Williams</i>		pop	61	[116]
<i>Uspop 2002</i>		pop	195	[28]
<i>RWC-Pop</i>		pop	100	[161]
<i>Schubert-Winterreise</i>		classical	225	[402]
<i>Weimar Jazz Database</i>		jazz	456	[309]
<i>JAAH</i>		jazz	113	[129]
Total			2240	

Table 6.1: *ChoCo partitions used for training and evaluating ChordSync.*

6.3. ChordSync: Conformer-Based Alignment of Chord Annotations to Music Audio

Method	Method	Genre	Precision \uparrow	Recall \uparrow	F1 Score \uparrow
<i>HCDF</i>		pop/rock	0.4999	0.6334	0.5269
<i>HCDF</i>		classical	0.4454	0.6220	0.5191
<i>HCDF</i>		jazz	0.4911	0.7749	0.5857
<i>HCDF</i>		all	0.4953	0.6508	0.5323
<i>ChordSync</i>		pop/rock	0.8847	0.8335	0.8553
<i>ChordSync</i>		classical	0.6008	0.5917	0.5951
<i>ChordSync</i>		jazz	0.4663	0.4129	0.4350
<i>ChordSync</i>		all	0.8895	0.8420	0.8621

Table 6.2: Precision, Recall, and F1 Score for the HCDF method [329] and the proposed ChordSync model.

of 2240 audio tracks, encompassing four distinct music genres: pop, rock, classical, and jazz. However, it is noteworthy to observe a significant imbalance in the dataset, with the pop/rock genre comprising over 65% of the total tracks.

Audio data is segmented into intervals of 15 seconds duration, with a 3-second overlap between each segment and the preceding one, yielding a corpus of 31909 segments. We split these segments into train, validation, and test sets with proportions of 65 – 20 – 15. Importantly, when a segment from a particular song is included in the train set, we ensure that no segments from the same song are included in either the validation or test sets.

Chord Changes Detection Evaluation

The first comparison is conducted with the Harmonic Change Detection (HCD) algorithm [180], which specialises in detecting harmonic changes on an audio signal. These algorithms are typically evaluated by assessing their capacity to detect the onsets of annotated chords within the identified harmonic changes, often employing standard metrics such as Precision, Recall, and F1 Score.

However, by their intrinsic design, HCD algorithms extract every harmonic variation present in the audio signal. [180] and [329] provide two distinct implementations of this algorithm, each optimising either the F1 score or precision. The number of harmonic changes varies significantly depending on the chosen algorithm implementation, but in general, it far exceeds that of chord changes.

In contrast, *ChordSync* extracts the number of chord changes of the list of chords passed to the CTC decoder. Table 6.2 presents a comparative analysis between the HCD algorithm in [329] and *ChordSync*. A harmonic change match is defined in a 0.3 seconds window between the predicted and the ground-truth onsets.

Chapter 6. Exploring Symbolic Limitations: Multimodal Strategies for Enhanced Harmonic Analysis

Our method demonstrates notable efficacy in chord change extraction, substantially increasing all the performance measures considered. This performance improvement stems from the model’s inherent design, which optimises the alignment between the audio signal and the provided sequence of chords. However, performance decreases in the less represented genres within the dataset, such as jazz and classical.

Alignment Evaluation

Method	Dataset	Percentage Correct \uparrow	Median Absolute Error \downarrow	Average Absolute Error \downarrow	Perceptual \uparrow
<i>DTW</i>	schubert-winterreise	0.8621	0.0661	0.2088	0.7895
<i>ChordSync</i>	schubert-winterreise	0.8245	0.2641	0.2512	0.7230
<i>ChordSync</i>	all	0.8664	0.4224	0.5001	0.7900

Table 6.3: *Performance of ChordSync on the Schubert-Winterreise dataset [402] compared to a standard DTW approach performed using the SyncToolbox library (first two rows). Additionally, performance metrics of the ChordSync method applied across all datasets are presented. Metrics are computed with the alignment metrics from the mir_eval library.*

Evaluating audio-to-score or audio-to-lyrics alignment entails comparing predicted and ground truth timestamps to measure their temporal differences [261, 145]. This comparison typically occurs pairwise and involves calculating metrics such as the median absolute error in seconds and the percentage of overlapping segments. This approach offers a straightforward means of assessing alignment accuracy and determining the effectiveness of alignment methods for practical applications.

Furthermore, perceptually-grounded metrics for evaluating lyrics-to-audio alignment systems have been recently introduced [257]. These metrics were fine-tuned on data collected through a user Karaoke-like experiment, reflecting human judgement of how “synchronous” lyrics and audio stimuli are perceived in that setup.

All the metrics described above are implemented in the `mir_eval` library [323], providing a standardised and accessible means for conducting evaluations in audio alignment. Given its similarities with other alignment tasks and the perceptual considerations involved, the same metrics are suitable for evaluating audio-to-chord alignment.

We compare the performance of *ChordSync* and a conventional DTW-based approach using the *SyncToolbox* library [288], which offers a diverse array of DTW-based implementations. The evaluation of this type of approach requires both symbolic sequences weakly aligned to audio, which are a prerequisite for the align-

ment, and ground truth annotations strong aligned to audio for evaluation. To our current knowledge, such annotations are exclusively found within the Schubert Winterreise dataset [402]. Consequently, the evaluation of this approach is constrained to a limited number of pieces and to the *classical* genre.

To perform the alignment between audio and chord annotations, the chord annotations were first decomposed into their constituent notes, each of which was then associated with the chord’s symbolic onsets. The audio data underwent pre-processing using chroma and DLNCO features, known for their effectiveness in alignment tasks [131]. Finally, alignment was carried out utilising memory-restricted multi-scale DTW (MrMsDTW) [285, 319].

Table 6.3 shows the performance of the proposed model on the Schubert Winterreise dataset compared to a standard DTW approach, along with the broader performance metrics of the ChordSync method applied across all datasets (c.f. Table 6.1). This evaluation demonstrates that the proposed model accurately detects chord changes and achieves alignment performance comparable to that of a DTW-based approach. Conversely, the evaluation conducted solely on a subset of the Winterreise dataset demonstrates performance comparable to DTW, albeit slightly lower. However, this data highlights the model’s strong generalisation capabilities, as it effectively aligns songs from a genre that was statistically rare in the training data due to its limited size.

Even so, it is worth noting that the proposed model achieves these results without relying on weak-aligned data, which is a requirement for DTW-based approaches.

6.4 Inter-annotator Agreement Analysis

As outlined in the introduction to this chapter, the metrics employed to evaluate inter-annotator agreement in chord datasets are constrained to binary distances [302]. The binary distance $B_{\text{dist}}(C_1, C_2)$ between two chords C_1 and C_2 equals 1 if $C_1 = C_2$, and equals 0 if $C_1 \neq C_2$.

Usually, the binary distance measure is weighted across by the duration of the annotation. This metric takes the name of Chord Symbol Recall (*CSR*) [179], and consists of weighting each distance (or similarity) by the duration of the corresponding segment:

$$CSR = \frac{|S_a \cap S_e|}{|S_a|}. \quad (6.5)$$

That is, the summed duration of time periods where the correct chord S_e has been identified, normalised by the total duration of the evaluation data S_a .

These evaluations are usually performed at different levels of granularity, such as root note, triads, inverted triads, tetrads, inverted tetrads, sevenths, inverted sevenths, and mirex chord metrics. An illustrative implementation of such evaluation metrics can be found in the chord module of `mir_eval` [323], a widely utilised toolkit for music information retrieval research.

However, assessing the musical content using a binary evaluation approach can potentially lead to misleading conclusions. In order to overcome this, new metrics have recently been introduced. One notable contribution in this regard is presented in [266], where the authors introduce three novel metrics designed to offer a more nuanced assessment of chord annotation performance: Spectral Pitch Similarity, Tone-by-Tone distance, and Mechanical Distance. Spectral Pitch Similarity is a measurement of the perceived pitch content of chords, grounded on psychoacoustic assumptions. On the other hand, Tone-by-Tone Distance considers each chord as a set of pitch classes, categorised as either tonal or neutral. This metric computes the proportion of pitch classes shared between each chord pair, culminating in a distance measure that reflects the overall similarity in pitch content. Lastly, Mechanical Distance offers a granular evaluation by treating chords as pitch class sets and approximating the physical distance between two chord labels as they would be played on an instrument. This metric extends the concept of Tone-by-Tone Distance by not only considering the proportion of incorrect pitches but also quantifying the magnitude of deviation for each erroneous note from the target chord, which by default is measured in semitones.

Additionally, we extend our study by combining the mechanical distance with a consonance-based distance. We use the consonance vector presented in [155], which is informed by perceptual studies tailored to Western tonal music analysis, and is defined as follows:

$$vt = [0, 7, 5, 1, 1, 2, 3, 1, 2, 2, 4, 6]. \quad (6.6)$$

Intervals such as perfect fifths and thirds (P5, m3, and M3) are regarded as the most consonant and are assigned the lowest value (value 1). Complementary intervals such as perfect fourths and sixths are deemed to be of intermediate consonance (value 2). Conversely, intervals ranked higher in dissonance, including major and minor sevenths, major seconds, major sevenths, and minor seconds, are

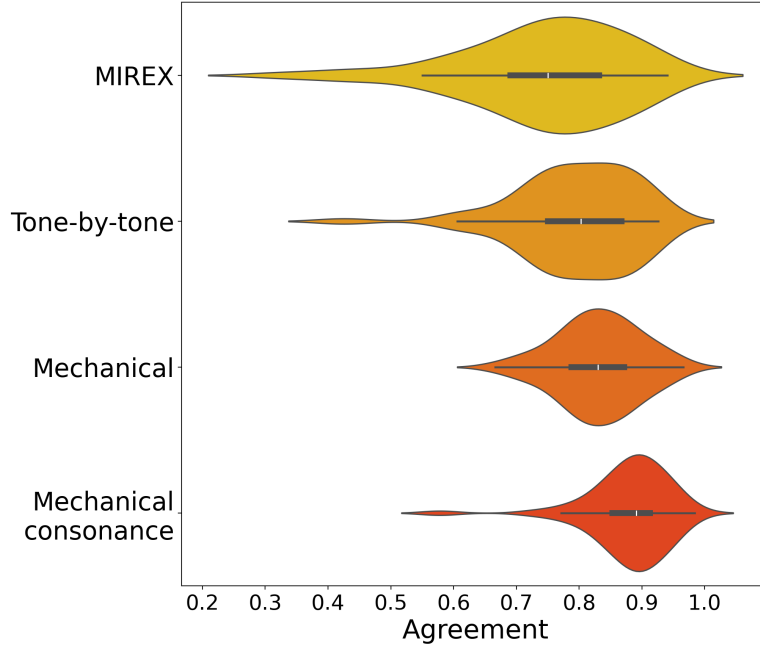


Figure 6.7: *Evaluation of Inter-Annotator Agreement using Chord Symbol Recall metrics across various distances: Binary (MIREX metrics), Tone-by-Tone, Mechanical, and Mechanical Consonance distances.*

assigned higher values (3, 4, 5, 6, and 7, respectively).

In this study, we compare our inter-annotator agreement analysis with the approach proposed in [227]. For the experiments, we use the same dataset as in the original evaluation: the Chordify Annotator Subjectivity Dataset [225], which is composed of 50 songs, each of which is annotated by four different music experts. In order to compare the metrics, they are normalised as follows: Tone-by-Tone distance T_{dist} is by design normalised between 0 and 1; hence the agreement measure is $1 - T_{\text{dist}}$. Conversely, mechanical distance M_{dist} returns non-normalised values, that we normalise by the maximum distance value obtainable with the distance vector being used (12 for semitone distance and 16 for the consonance distance):

$$\text{agreement}_{M_{\text{dist}}} = 1 - \frac{M_{\text{dist}}}{\max(M_{\text{dist}})} \quad (6.7)$$

We compute the CSR for each of the metrics at different granularity levels, encompassing the metrics normally used in ACE evaluation. Table 1 reports the results of the evaluation as the average for all the combinations for all of the 50 song of the dataset, while Figure 6.7 depicts the distributions of all experiments on complete chords.

	Binary	TbT	MD	MDC
Root	0.757	0.757	0.929	0.960
Majmin	0.734	0.952	0.987	0.990
Thirds	0.741	0.945	0.981	0.986
Triads	0.712	0.940	0.980	0.983
Tetrads	0.572	0.942	0.972	0.979
Sevenths	0.592	0.950	0.979	0.984
MIREX	0.744	0.794	0.831	0.879

Table 6.4: Comparison of Inter-Annotator Agreement Evaluation Metrics. The table presents the evaluation results of inter-annotator agreement using various metrics, including binary metrics proposed in [227], Tone-by-Tone distance (TbT), Mechanical Distance (MD), and Mechanical Distance with Consonance (MDC). The MIREX row computes the MIREX metric for binary distance and chord-to-chord distances for all the other metrics.

The results of our evaluation reveal a notable improvement in agreement when employing Tone-by-Tone and Mechanical distances compared to binary distance. Specifically, mechanical consonance-based scores demonstrate higher levels of agreement overall and tends to yield lower agreement scores than its semitone-based counterpart. This observation suggests that annotators converge more consistently on the fundamental harmonically consonant structures when transcribing chords.

6.5 From Dissonance to Harmony

6.5.1 Methods

As a main contribution of this section, we propose a novel model for ACE. The novelty encompasses three key aspects: (i) we present a novel architecture based on the conformer architecture tailored specifically for ACE; (ii) we introduce a consonance-based smoothing technique applied to the target labels, improving the

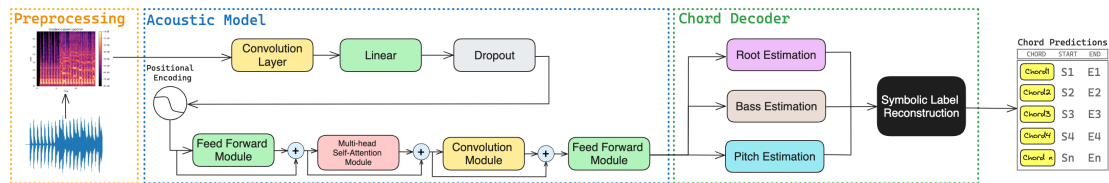


Figure 6.8: Overview of the Model Architecture, which comprises the preprocessing stage, the conformer-based model, and the symbolic chord decoder.

model’s training by integrating perceptual characteristics of harmony; and (iii) we present an enhanced version of the chord label encoding/decoding methodology initially introduced in [263].

Proposed Model

The proposed model, illustrated in Figure 6.8, leverages the conformer architecture [170], which, combining the strengths of CNNs and self-attention mechanisms, captures local and global dependencies within sequences.

The audio is first resampled to a sampling rate of 22050 Hz, and a hop size of 2048 is applied. Then, the CQT features are calculated on 6 octaves starting from $C1$, with 24 bins per octave, resulting in a total of 144 bins. The CQT features are fed to a convolutional block, which includes batch normalisation, convolutional and dropout layers. The output of the convolutional block is forwarded to a conformer encoder consisting of 16 blocks [170] before being passed to the decoder layers.

Label encoding follows a similar approach as [263]. Specifically, root and bass notes are encoded as a 13-dimensional one-hot vector, with the first 12 positions representing the 12 semitones from C to B , and the thirteenth position indicating silence (denoted as N). Conversely, chord pitches are encoded using a 12-dimensional multi-hot encoding scheme. Each dimension corresponds to a musical pitch, and the values range between 0 and 1, representing the activation of individual notes within the chord.

The output of the conformer layers is fed into three distinct fully-connected layers dedicated to predicting bass, root, and pitch activations, respectively. These fully-connected layers transform the high-dimensional representations learned by the conformer into probability distributions over the possible values for each label component. The output layer for predicting bass and root notes uses a *softmax* activation function, while the output layer for predicting chord pitches uses a *sigmoid* activation function.

Consonance-based Label Smoothing

We present a novel consonance-based smoothing of the target labels to train the network according to the formalised concepts of consonance discussed in Section 6.4.

Label smoothing is commonly formulated using the cross-entropy loss function

with soft targets. Let's denote the ground truth label for a given example as y_i , where y_i is a one-hot encoded vector representing the true class, and let \hat{y}_i denote the predicted probability distribution outputted by the model for the same example. The cross-entropy loss function \mathcal{L} can be expressed as:

$$\mathcal{L}(y_i, \hat{y}_i) = - \sum_{j=1}^C y_{ij} \log(\hat{y}_{ij}) \quad (6.8)$$

where C is the total number of classes, y_{ij} is the j -th element of the ground truth label vector y_i , and \hat{y}_{ij} is the predicted probability of the j -th class by the model.

With label smoothing, instead of using hard targets (one-hot encoded vectors), soft targets \tilde{y}_i are introduced as a combination of the ground truth label y_i and a uniform distribution over the classes, which are calculated as:

$$\tilde{y}_{ij} = (1 - \epsilon) \cdot y_{ij} + \frac{\epsilon}{C} \quad (6.9)$$

where ϵ is a smoothing parameter (typically a small value), and \tilde{y}_{ij} is the j -th element of the soft target vector \tilde{y}_i .

The smoothing is calculated in the LS paradigm according to the consonance vector vt presented in Equation 6.6. Let us denote the smoothing value for the i -th index as s_i . We can calculate s_i as follows:

Once we obtained the smoothing values, we can normalise them to obtain the soft labels. Let $\tilde{\mathbf{y}}$ represent the soft label vector, and \mathbf{s} represent the smoothed values vector. Then, the soft label \tilde{y}_i for the i -th index can be calculated as:

$$\tilde{y}_i = \frac{s_i}{\sum_{j=1}^C s_j} \quad (6.10)$$

C being the total number of classes.

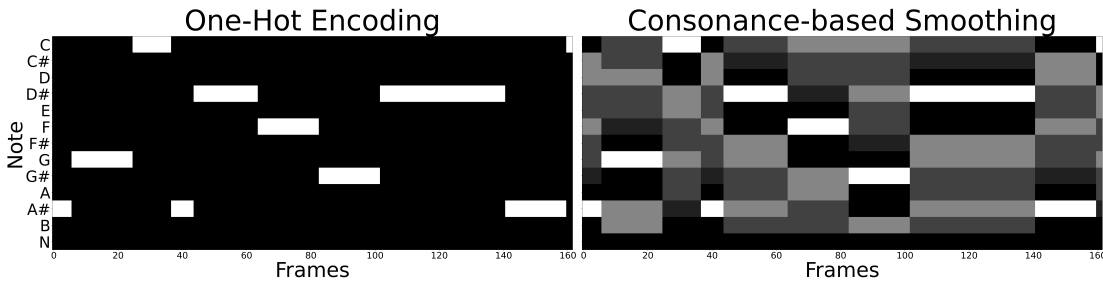


Figure 6.9: Comparison of the one-hot encoding of the root note (left) with the root note encoding after applying the proposed consonance-based smoothing (right).

Figure 6.9 exemplifies the smoothing proposed, comparing it to hard labels. Consonance-based smoothing is applied to all targets, ensuring the soft labels reflect the consonance relationships among different chord components.

We use binary cross-entropy loss for chord activations, which is well-suited for binary classification tasks. Conversely, to maximise the effect of the smoothing for root and bass classification, we use focal loss [246], a specialised loss function designed to address class imbalance by down-weighting well-classified examples during training.

Chord Decoding

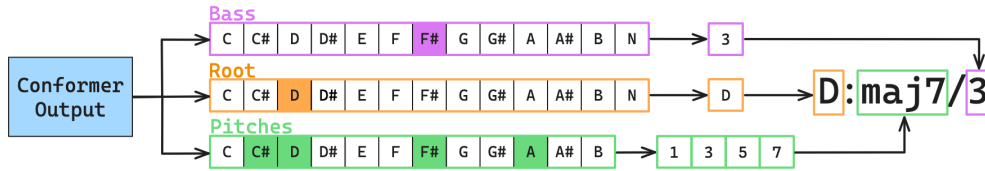


Figure 6.10: Example of chord label decoding for a *D:maj7/3* chord using the architecture from [263].

The predicted probabilities are decoded as follows. First, we identify the root note by selecting the highest probability and then decode its numerical representation into the corresponding pitch class symbol. A threshold is set for the pitch activation probabilities (by default, at 0.5), and only activations surpassing this threshold are considered. To maximise the recognition of complete chords, we iteratively lower the threshold by 0.1 until a chord with at least 2 notes is recognised, while we lower it by 0.1 only once if the chord is composed of only two notes. Subsequently, pitches are converted to intervals by calculating the distance from the predicted root note to each predicted pitch. The same procedure is applied for the bass note, enabling the reconstruction of the chord, as depicted in Figure 6.10. Ultimately, the reconstructed chord is passed to the `harte_library`⁶, which offers utilities for converting the predicted chord label into the respective shorthand notation.

6.5.2 Evaluation

In this section, compare the performance of the proposed ACE model with a state-of-the-art method [301], using both standard metrics and mechanical dis-

⁶`harte_library`: <https://github.com/andreamust/harte-library>

Chapter 6. Exploring Symbolic Limitations: Multimodal Strategies for Enhanced Harmonic Analysis

tances. We also explore the learned representations to assess the effectiveness of the proposed consonance-based smoothing.

For all the experiments we use a subset of ChoCo [90], filtering the partitions contained in the corpus and selecting the ones that contain time information expressed in real time (seconds), and that are of genre *pop* or *rock*. All datasets used are listed in Table 6.5. We split each track into 15-second excerpts with a 2-second overlap. We also ensure that excerpts from the same track are not shared among different sets. We employ data augmentation by transposing both audio and targets from -5 to $+6$ semitones. During training, we use the *AdamW* optimiser and cosine annealing with warm restart learning rate scheduler to dynamically adjust the learning rate during training cycles. Additionally, we adopted mixed precision training [274] to accelerate training. To prevent overfitting, we implement early stopping, terminating training when performance on a validation set ceased to improve after 5 epochs.

Evaluation of the ACE Model

We evaluate our model using mechanical distance and mechanical distance with consonance, as detailed in Section 6.4, alongside standard `mir_eval` binary metrics. The experimental results, as summarised in Table 6.6, reveal that our model demonstrates comparable performance to the BTC model [301], widely regarded as state-of-the-art in audio chord estimation. As documented in [211], minimal differences exist among models when using standard metrics. However, our proposed model demonstrates consistent improvements in performance compared to state-of-the-art models across `mir_eval` metrics. Moreover, our model performs better than the BTC model in predicting complex chords, such as tetrads and sevenths. Most notably, our model significantly outperforms the BTC model in terms of mechanical consonance, highlighting its effectiveness in capturing har-

DatasetDataset	Genre	#Tracks#Tracks	Reference
<i>Isophonics</i>	pop, rock	300	[258]
<i>Billboard</i>	pop	740	[48]
<i>Chordify</i>	pop	50	[225]
<i>Robbie Williams</i>	pop	61	[116]
<i>Uspop 2002</i>	pop	195	[28]
<i>RWC-Pop</i>	pop	100	[161]
Total		1446	

Table 6.5: *ChoCo partitions used for training the Audio Chord Estimation model.*

6.5. From Dissonance to Harmony

Model	Smoothing	Root	MajMin	Triads	Tetrads	7th	MIREX	MD	MD-C
Ours	None	0.869	0.811	0.798	0.681	0.699	0.842	1.690	1.454
Ours	Linear	0.871	0.819	0.803	0.703	0.683	0.812	1.357	1.241
Ours	Consonance	0.863	0.837	0.823	0.755	0.730	0.872	1.264	1.065
BTC [301]	None	0.827	0.848	0.786	0.663	0.655	0.808	2.176	1.991

Table 6.6: Performance comparison between the Conformer architecture proposed, evaluated using different types of label smoothing. Moreover, performances are compared with the bi-directional transformer architecture proposed in [301].

monic relationships. Furthermore, consonance-based smoothing proves to perform generally better than linear smoothing, tested with $\alpha = 0.1$. Finally, both mechanical distance and mechanical distance with consonance exhibited notably superior performance on the model trained with consonance-based smoothing.

Penultimate Layer Representation

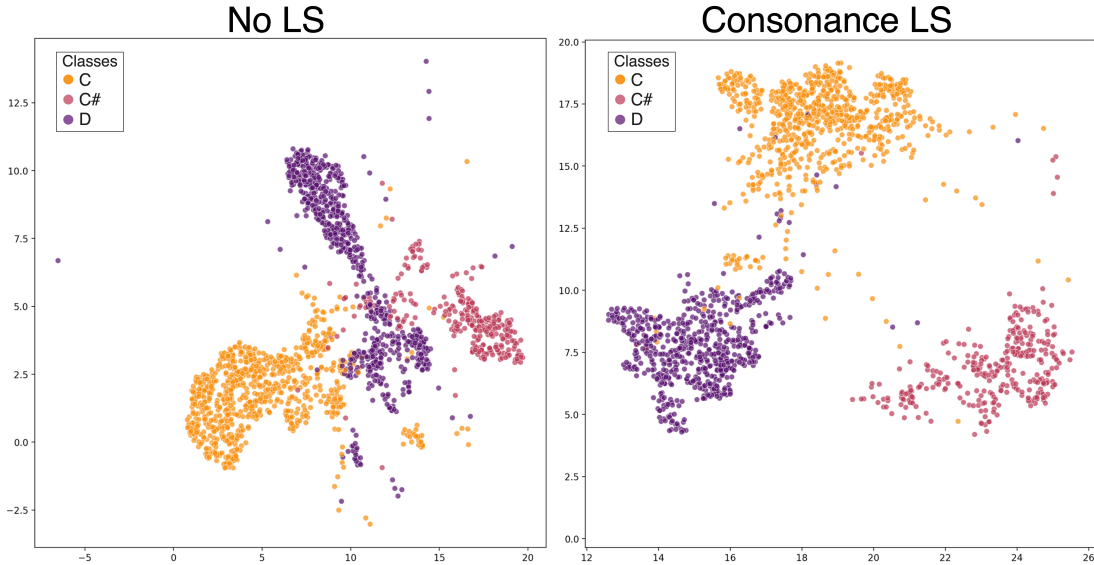


Figure 6.11: Penultimate layer representation [287] of notes C , $C\#$, and D training without smoothing (left) and with consonance-based smoothing (right).

In [287], authors demonstrate that label smoothing encourages activations in the penultimate layer to align closely with the template of the correct class while maintaining equidistance from templates of incorrect classes. Inspired by these findings, we conduct similar experiments on the penultimate layer of our proposed conformer model. Specifically, we compared the penultimate layer representations of the conformer model both without and with consonance smoothing. Our visualisation experiment follows the same approach of [287]: 1. Selection of three

classes of interest; 2. Determination of an orthonormal basis for the plane intersecting the templates of these three classes; and 3. Projection of penultimate layer activations from examples of these three classes onto this plane. Using UMAP [264] for dimensionality reduction of the penultimate layer features, our visualisation depicts how activations cluster around the templates in a 2-D representation and demonstrates how label smoothing enforces a structured distance relationship between examples and clusters from other classes.

This analysis, depicted in Figure 6.11, provides insights into how consonance smoothing influences the representation of chord classes in the penultimate layer, shedding light on the model’s ability to capture harmonic relationships. Notably, the visualisation reveals significantly tighter clusters, indicating that consonance-based smoothing promotes equidistance of each example in the training set from all templates of other classes.

6.6 Conclusion

In this chapter, we introduced two major contributions aimed at addressing central challenges in harmonic data analysis through multimodal approaches.

First, we presented *ChordSync* (Section 6.3), a novel model for audio-to-chord alignment based on the Conformer architecture [170]. *ChordSync* achieves performance comparable to traditional Dynamic Time Warping (DTW) algorithms but eliminates the need for pre-existing weak alignment, streamlining the process of synchronizing chord annotations with audio. By leveraging resources such as crowd-sourced chord annotations, which often lack timing information, *ChordSync* supports the creation of diverse and comprehensive audio-aligned chord datasets, with demonstrated practical utility. To encourage reuse, we provide a pre-trained model and a user-friendly library, enabling easy synchronization of chord annotations with audio.

Building on this multimodal dataset, we proposed an innovative Audio Chord Estimation (ACE) model, also utilizing the Conformer architecture, and incorporating a consonance-based label smoothing approach. This method was inspired by our analysis of inter-annotator agreement, which underscored the importance of consonance and dissonance concepts in managing discrepancies between annotators. By embedding consonance-informed labels into the model’s learning process, we enhanced its capacity to capture subtle harmonic relationships, particularly in

cases where traditional binary metrics fall short.

To address chord vocabulary imbalance, we introduced a chord decoder that separately predicts root note, bass note, and chord notes activations, reducing reliance on extensive vocabularies and improving accuracy for uncommon chords. The proposed approaches demonstrate improvements in the learned harmonic representation, opening new pathways for refined harmonic analysis and further applications in MIR.

6.6.1 Limitations and Future Work

While the contributions of this chapter represent substantial advancements, both *ChordSync* and the proposed Audio Chord Estimation (ACE) model have certain limitations that offer directions for future research.

ChordSync. Despite its effectiveness, ChordSync relies on a simplified chord vocabulary, which constrains its performance. Chords that are absent from the model’s vocabulary are approximated by the nearest available chord, which may result in inaccuracies, particularly in scenarios where consecutive chord symbols are identical. This limitation can hinder the CTC decoder’s ability to maintain precise alignment. Additionally, ChordSync’s model is not key-agnostic, introducing potential alignment discrepancies when there are key differences between the chord labels and the audio signal. Future work should consider exploring alternative chord encodings and methods for achieving key independence, which would enhance the model’s flexibility and accuracy. Another avenue for improvement is exploring novel architectures that extend beyond the acoustic model paradigm, such as utilizing CTC as a loss function within a semi-supervised learning framework, which could further enhance model generalizability and efficiency.

ACE Model. The proposed ACE model was trained on a subset of the ChoCo dataset due to limitations in available GPU resources, which restricted the scale of training. Expanding this model to incorporate the full ChoCo dataset in future work would likely yield more robust harmonic representations. Additionally, we demonstrated that the representations learned by the model benefit from consonance-based label smoothing, offering an improvement over standard ACE models that do not incorporate this nuanced approach to harmonic relationships.

Future work may explore the application of this model to a variety of tasks beyond audio chord estimation. For instance, the model could be employed to

refine and harmonize existing chord annotations across different corpora, reducing annotator subjectivity and creating more consistent chord representations. Furthermore, the learned representations could support tasks such as cover song detection and version identification from audio, areas where symbolic-informed models have shown promise [418]. Unlike previous methods that only consider root notes, our model’s detailed harmonic profile suggests it could perform well in these contexts, leveraging its comprehensive harmonic understanding to enhance accuracy and consistency across tasks.

CHAPTER 7

Conclusion

This thesis addresses fundamental challenges in MIR by advancing KR and ontology engineering, specifically tackling issues of data fragmentation, interoperability, and scalability in music data. Through the development of a unified semantic model (c.f. Chapter 3), we respond to the need for consistent and flexible frameworks that can integrate diverse musical representations and datasets across the field.

Building on this model, we create a workflow for integrating symbolic datasets and apply it to produce a large, harmonized corpus of harmonic annotations (c.f. Chapter 4). This corpus exemplifies the utility of data integration and the potential for more structured and interconnected music datasets within MIR.

Moreover, we demonstrate how data integration can enrich both musicological exploration and creative processes (c.f. Chapter 5). Using the large, harmonised corpus and developing novel local similarity functions, we uncover new insights into musical structure, style, and evolution, providing a robust foundation for analysis across diverse genres and historical periods. This integrated data also supports creative applications, enabling tools that assist musicians and composers

by offering inspiration drawn from a broad spectrum of harmonic structures and styles.

Finally, we address the limitations of data integration in harmonic analysis, especially regarding the inherent subjectivity in symbolic harmonic annotations (c.f. Chapter 6). To address this challenge, we embrace a multimodal approach by enhancing our dataset with aligned audio. By integrating audio and symbolic data, we aim to create harmonic representations from audio that transcend the subjectivity inherent in individual human annotations.

7.1 Summary of Contributions

In this section, we summarise the contributions of this thesis by reviewing the research questions presented in Chapter 1 and discussing the solutions formulated for each question.

RQ1

The first challenge addressed in this thesis is the fragmentation and lack of interoperability in music data, which impedes the integration of diverse musical datasets and complicates the replicability of research outcomes, as identified in RQ1 (c.f. Section 1.1.1). This fragmentation results from the absence of a unified approach to music representation, as music data involves various dimensions that require flexible descriptions. These dimensions encompass metadata (context) and content, such as symbolic annotations and audio, each presenting unique challenges in standardisation and interoperability.

To address these issues, this thesis introduces the Polifonia Ontology Network (PON), an ontology framework which comprises 15 new ontologies, designed to unify representations of metadata, annotations, performance mediums, and historical sources (c.f. Chapter 3). By formalising the semantics of both musical content and context, PON enables the creation of interoperable KGs from diverse music datasets, addressing core fragmentation challenges in MIR.

Furthermore, this work contributes to the field of Knowledge Engineering (KE) by releasing an extension of the eXtreme Design (XD) methodology and a comprehensive dataset of Competency Question (CQ) that guide music ontology development and testing.

Finally, we provide evidence of current and potential reuse by institutional

and industrial stakeholders, demonstrating the practical relevance and impact of PON.

RQ2

This thesis directly addresses **RQ2** (c.f. Section 1.1.2) by tackling the specific fragmentation challenges within harmonic data—a domain that exemplifies issues of data diversity due to its varied notational conventions and formats. To tackle these complexities, we developed the *Chord Corpus (ChoCo)* (c.f. Chapter 4), which serves both as a comprehensive resource and a structured workflow aimed at standardising and integrating symbolic datasets with heterogeneous formats and annotation practices.

The ChoCo workflow applies the ontology model proposed in **RQ1** to harmonise over 20,000 high-quality harmonic annotations from 18 distinct chord datasets. By using the JAMS data structure as a unified annotation model, ChoCo achieves interoperability across three levels: metadata, annotation format, and chord notation.

The resulting ChoCo KG, with its ≈ 30 million RDF triples and 4,000+ links to external datasets, supports advanced semantic analysis and facilitates large-scale musicological and computational studies. This resource not only promotes accessibility across various musical traditions and genres but also exemplifies how the ontology model developed in **RQ1** can be applied to resolve fragmentation in a complex use case. A survey conducted with potential users from the MIR and SW communities also highlighted strong interest in adopting both the ChoCo dataset and workflow, underscoring its relevance and potential impact.

RQ3

The creation of large, harmonized corpora of symbolic harmonic annotations introduces unprecedented opportunities to explore harmonic data at scale, directly addressing **RQ3** (c.f. Section 1.1.3). By integrating a variety of datasets, this work enables detailed analysis across a broad spectrum of musical genres, styles, and artist collections, providing new insights into harmonic content previously constrained by fragmented data sources.

A structured approach to similarity is essential for fully leveraging these harmonized corpora, enabling in-depth analysis and interpretation of harmonic content across large datasets. Unlike traditional methods that rely on general metadata

or surface-level features, content-based harmonic similarity delves into the core of musical patterns and relationships. This deeper analysis reveals nuanced harmonic structures and recurring patterns that help define genres, trace stylistic shifts, and capture the distinctive characteristics of individual artists or historical periods.

To this end, we propose and evaluate two key contributions in this area: *LHARP* and *Harmory*.

LHARP is a novel method that identifies local harmonic similarities across pieces, focusing on detecting recurring harmonic patterns within musical sequences. Unlike traditional global similarity measures, which capture broad stylistic or structural similarities, LHARP provides a more flexible and interpretable similarity measure tailored for musicological tasks that require detailed harmonic comparisons across works (c.f. Section 5.3).

Harmory, the Harmonic Memory, is a KG designed to assist computational creativity by organizing harmonic patterns within a structured, musically meaningful space. Built on cognitive and musicological principles, Harmory captures both temporal relationships and structural similarities between harmonic sequences, linking them within a broader harmonic landscape. To do this, we propose and evaluate two novel state-of-the-art algorithms for both harmonic segmentation and harmonic similarity. This resource enables transparent and accountable access to diverse harmonic structures, offering musicians and composers a valuable foundation for compositional assistance, where they can explore and experiment with harmonic ideas across genres and styles (c.f. Section 5.4).

Through these contributions, this thesis demonstrates the potential of harmonized corpora in advancing both musicological research and creative applications, underscoring the value of large-scale data integration in supporting complex musical inquiries and enhancing creative workflows.

RQ4 and RQ5

While in Chapters 5 we demonstrate how symbolic data integration can advance standard MIR tasks, particularly in the domain of harmonic similarity, the process of data integration and corpus creation also brings to light several limitations that still impede the full realization of data-driven approaches in music analysis, as highlighted in RQ4 and RQ5. These limitations underscore the inherent challenges of working with diverse and subjective harmonic annotations, as well as

the imbalance across harmonic datasets.

To address these issues, we first make our dataset multimodal by introducing *ChordSync* (Section 6.3), a novel audio-to-chord alignment approach that leverages the Conformer architecture [170], hence addressing RQ4 (c.f. Section 1.1.4). ChordSync provides an efficient solution for synchronizing symbolic chord annotations with audio by eliminating the requirement for pre-existing weak alignments, achieving performance comparable to traditional Dynamic Time Warping (DTW) algorithms. This model enables the transformation of crowd-sourced chord annotations, which often lack precise timing information, into cohesive multimodal datasets. To promote accessibility and reproducibility, we release a pre-trained model along with an easy-to-use library, allowing researchers and practitioners to synchronize chord annotations with audio across varied datasets.

Building on this multimodal dataset, we introduce an advanced Audio Chord Estimation (ACE) model, also based on the Conformer architecture, to address the challenges outlined in RQ5 (c.f. Section 1.1.5). This model integrates a consonance-based label smoothing technique, developed to handle the inconsistencies commonly observed in inter-annotator agreements. By embedding consonance-informed labels into the training process, the model enhances its ability to capture nuanced harmonic relationships, effectively accommodating the subjectivity and diversity of chord annotations. Additionally, our model includes a chord decoder that separately predicts root note, bass note, and chord note activations, which enhances its adaptability and accuracy, particularly for uncommon chords. These innovations not only improve transcription accuracy but also pave the way for applications that harmonize and refine symbolic chord annotations, reducing dependence on the annotators’ subjective interpretations.

7.2 Discussion and Future Work

We conclude this thesis with a discussion of potential directions for future work, some of which are already in progress.

7.2.1 Ontology Engineering and Data Integration

As a next step in ontology engineering, we plan to conduct a comprehensive competency question-driven evaluation of PON’s modules to further refine and validate the ontology’s structure and content. This evaluation will help ensure that

PON effectively addresses the practical requirements of diverse musicological and computational use cases. Additionally, we aim to support stakeholders and early adopters in reusing, extending, and maintaining the ontologies and knowledge graphs over the long term.

Furthermore, we plan to extend PON modules, for instance by incorporating novel music theories in the Music Analysis Module. Lastly, we aim to enhance interoperability by aligning PON with other relevant ontologies, such as those discussed in Section 3.2.

7.2.2 Dataset Expansion and Workflow Adaptation

Future work on dataset development will focus on expanding *Chord Corpus (ChoCo)* by integrating additional harmonic data, thereby increasing its scope and enhancing its utility for a wider range of applications. Beyond harmonic data, the standardized workflow introduced in this thesis can be applied to create new corpora encompassing various types of music annotations. In fact, ChoCo’s workflow and the JAMS ontology have already been applied to represent a corpus of melodic pattern data [353]. Building on this initial success, the workflow could be adapted to generate additional corpora that focus on melodic, structural, or rhythmic annotations, thereby enriching the resources available for musicological research and computational analysis.

7.2.3 Symbolic Harmonic Similarity and Exploration

While LHARP demonstrates promising potential for generalizing to various symbolic sequences with structural similarities to music, several limitations remain, suggesting directions for future work. A primary concern involves the data preprocessing and encoding steps. Specifically, the method of transposing all pieces to a common key, while practical for similarity calculations, may raise concerns among music experts who argue that it could alter musical texture, potentially compromising the integrity of the original (non-transposed) version. Additionally, the encoding method, which decomposes chords into individual pitches, may overlook context-specific nuances—such as when two distinct chord labels produce the same sound but possess unique harmonic functions that are lost in pitch-based encoding. Although these simplifications offer practical advantages, gathering expert feedback remains essential for validating their effectiveness and exploring further refinements.

Some of these challenges are partially addressed by Harmory’s similarity function, which leverages DTW to enhance sequence alignment. However, Harmory’s encoding still depends on TPS—a method that, despite its effectiveness and musicological foundation, lacks context-awareness in representing musical harmony. Future research could investigate advanced representation learning techniques to achieve more contextually aware and robust harmonic encodings [275, 113]. Such techniques could capture harmonic nuances that current encodings overlook, resulting in a more expressive and precise measure of similarity.

Currently, LHARP and Harmory provide single measures of similarity, and their effectiveness is therefore naturally limited to the specific tasks and use cases evaluated. Expanding the experimental framework to include a broader set of tasks will be essential for assessing the general applicability of these methods. Furthermore, future work will involve developing new similarity functions that can better adapt to diverse musicological and creative applications, thereby enhancing the versatility of both LHARP and Harmory in supporting a wider array of music analysis and composition tasks.

7.2.4 Multimodality

While the contributions of this thesis mark substantial advancements in multimodal approaches for MIR, both *ChordSync* and the proposed Audio Chord Estimation (ACE) model exhibit certain limitations, which open pathways for future research and development.

ChordSync. Although ChordSync proves effective in aligning chord annotations with audio, it relies on a simplified chord vocabulary, which can restrict its performance. Chords absent from the model’s vocabulary are approximated to the nearest chord available, which can lead to inaccuracies, especially in cases where consecutive identical chord symbols pose alignment challenges. This limitation can impede the CTC decoder’s ability to achieve precise synchronization. Furthermore, the current model is not key-agnostic, which can introduce alignment discrepancies when the key of the chord labels differs from that of the audio signal. Future research could explore alternative chord encodings and develop methods for achieving key independence, enhancing both model flexibility and alignment precision. Another promising direction involves exploring architectures that transcend the acoustic model paradigm; for instance, utilising CTC loss function within a semi-supervised framework, which may improve model efficiency.

ACE Model. The ACE model, due to limited GPU resources, was trained on only a subset of the ChoCo dataset, thereby restricting the scale of training. Future work should involve expanding this model to incorporate the full ChoCo dataset, potentially resulting in even more robust harmonic representations. The consonance-based label smoothing used in this model demonstrated improvements over traditional ACE models by capturing subtleties in harmonic relationships that standard methods often overlook.

Further explorations could examine the application of this model to a range of tasks beyond audio chord estimation. For instance, it may be employed to refine and harmonize existing chord annotations, reducing annotator subjectivity and yielding more consistent harmonic representations across corpora. Additionally, the model’s learned representations could support tasks such as cover song detection and version identification, domains in which symbolic-informed models have shown considerable promise [418]. Unlike previous methods that primarily consider root notes, the detailed harmonic profile of our model positions it as a potentially effective tool for such tasks, offering an advanced harmonic understanding that enhances accuracy and consistency across diverse applications.

Acknowledgements

This research was funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004746.

I would like to express my sincere gratitude to my supervisor, Prof. Valentina Presutti, for her support, and for the opportunity to pursue this doctoral research.

I am deeply grateful to Dr. Jacopo de Berardinis for his mentorship during all stages of my PhD, which provided me with essential research methodologies and ethical foundations for academic work.

My sincere appreciation extends to all members of the Polifonia consortium, whose collaborative spirit and diverse expertise have enriched this research endeavour. Their contributions have been vital to the interdisciplinary nature of this work.

Special thanks go to the Music Technology Group (MTG) at Universitat Pompeu Fabra, Barcelona, where I spent a valuable research period. I am particularly indebted to Dr. Martín Rocamora for his supervision and mentorship which significantly contributed to this work.

List of Figures

2.1	Music score illustrating various enharmonic intervals, highlighting equivalent pitch pairs.	24
2.2	Visualization of the C major scale, displayed on both a piano keyboard and a musical staff, illustrating the corresponding notes across both representations and scale degrees.	25
2.3	Scale degrees and their corresponding names within both major (C major) and minor (C minor) scales.	26
2.4	Illustration of various intervals showing diminished, minor, major, augmented, and perfect qualities.	27
2.5	Illustration of relative major and minor keys using the circle of fifths, showing their interrelationships based on key signatures. . .	28
2.6	Recurring rhythmic motive from Chopin's Mazurka in G Minor, Op. 67, No. 2, illustrating the use of rhythmic patterns to create cohesion across the piece.	30
2.7	Illustration of phrasing in Mozart's Sonata in D Major, K. 284, I, highlighting the structural and thematic development across measures 1-4.	31
2.8	Examples of triads: augmented, major, minor, and diminished, illustrating the distinct interval structures that define each type. . .	35
2.9	Examples of chord extensions: Illustration of a major triad and its progressive extensions to seventh, ninth, eleventh, and thirteenth chords, built on the same root note.	36

List of Figures

2.10	Illustration of a basic $I - V - I$ harmonic progression, showing the movement from tonic to dominant and back to tonic, creating a sense of resolution.	37
2.11	Audio features extracted from the first 15 seconds of “Do I Wanna Know?” by Arctic Monkeys.	51
2.12	Symbolic formats of a B5 note in Eb with a time signature of 3/4: (a) score, (b) LilyPond, (c) **kern, (d) ABC, (e) GUIDO, (f) MusicXML, (g) MEI.	53
3.1	Summary of the main Polifonia extensions to the eXtreme Design methodology.	80
3.2	Visualisation of the Polifonia CQ embeddings using TensorBoard. .	81
3.3	Overview of the main modules in the Polifonia Ontology Network and its foundational models, with pilots as early adopters.	84
3.4	Graffoo example of ‘Highway Star’ by Deep Purple using 5 PON modules to describe metadata, instruments, and musical content annotations on two audio recordings, demonstrating interoperability despite differences in annotation formats.	88
3.5	Graffoo notation describing music artists as musicians, ensembles, and algorithms.	91
3.6	Graffoo diagram abstracting music inception as a creative process involving artists in activities, defined by time, space, and roles. . .	92
3.7	Graffoo diagram describing a music entity and its elements: Text, AbstractScore, and Instrumentation.	93
3.8	Graffoo diagram describing performance, recording, broadcasting, publication, and licensing.	94
3.9	Graffoo diagram illustrating the pattern to describe provenance with RDF*.	95
3.10	Selection of questions assessing respondents’ interest in adopting music ontologies, with responses measured on a Likert scale from 1 to 5.	98
4.1	Overview of the data transformation workflow for symbolic music annotations, used for chord and key annotations in the creation of the ChoCo Knowledge Graph.	109

4.2	Example of a harmonic progression annotated using different notation systems.	111
4.3	Overview of the Chonverter workflow, describing how different chord notations are converted to the Harte notation.	117
4.4	Graffoo diagram depicting a fragment of the JAMS ontology, illustrating JAMS files, provenance, musical objects, and annotations. .	121
4.5	Graffoo diagram illustrating a fragment of the JAMS ontology, focusing on JAMS annotations and observations, with time information handling highlighted.	122
4.6	Example of data modelled using the JAMS Ontology, extracted from a track in the Wikifonia dataset, with annotations and observations containing time references in beats and measures.	123
4.7	Graffoo diagram of the Roman Chord Ontology, describing Roman Numeral Chords and their constituting elements.	125
4.8	Example of a Knowledge Graph generated using the Roman Chord Ontology.	126
4.9	Distribution of audio track and score durations in the ChoCo dataset.	127
4.10	Overview of the ten most common performers and composers in the ChoCo dataset.	127
4.11	Distribution of the number of chord observations per annotation and their distinct chord classes in the ChoCo dataset.	130
4.12	Distribution of chord durations for audio and symbolic annotations in the ChoCo dataset.	130
4.13	Absolute and relative occurrences of chord classes in the ChoCo dataset, before and after removal of consecutively repeated chords.	130
4.14	Overview of survey responses to Questions 2 (music domains) and 3 (data types), assessing potential adoption of ChoCo.	139
4.15	Overview of survey responses to Questions 4-11, assessing background, relevance, and interest in working with chord data.	140
5.1	Working example of LHARP’s workflow with harmonic progressions from ‘Crazy Little Thing Called Love’ by Queen and ‘P.S. I Love You’ by The Beatles.	155
5.2	Distribution of each group/dataset on node-specific network metrics: degree, clustering coefficient, and closeness centrality in LHARP’s workflow, accompanied by post-hoc statistical analysis results. . .	160

List of Figures

5.3	Illustration of how nodes/tracks are distributed across the three communities in LHARP’s similarity results.	162
5.4	Illustration of the harmonic graph, encoding all the harmonic dependencies between tracks in the three music collections, using LHARP.	163
5.5	Overview of the main steps for creating Harmory, from encoding chord progressions in Tonal Pitch Space (TPS) and segmentation, to identifying harmonic patterns and generating the knowledge graph.	164
5.6	Example of a Harmonic SSM generated in Harmory from the TPS signal of ‘Crazy Little Thing Called Love’ by Queen, showing four main block-like structures that correlate with the musical form, along with smaller nested harmonic patterns.	167
5.7	Graffoo diagram illustrating the Harmory Ontology.	170
5.8	Structural coverage of known patterns for each segmentation method tested in Harmory.	173
5.9	Example of a generated chord progression using a pattern-based prompt. Given a first segment, each segment is chosen according to similarity to the subsequent one in the original sequence, and filtered according to arbitrary criteria.	175
6.1	Distribution of chord types in the ChoCo dataset.	183
6.2	Basic schema of ChordSync: The model processes a list of chords alongside the audio signal, producing time-aligned chords as output.	192
6.3	Architecture of ChordSync: audio preprocessing to Constant-Q Transform, conformer-based acoustic model training, and CTC forced alignment for chord label output.	193
6.4	Data augmentation policies for ChordSync, including random masking of the CQT along the time axis, frequency axis, or both.	194
6.5	Workflow of the pre-processing applied to chord labels in ChordSync, where labels are numerically encoded and upsampled to match the length of the CQT.	196
6.6	Example of chord classes predicted by the conformer-based acoustic model in ChordSync, with the upper panel showing target labels and the lower panel displaying predicted probabilities.	197

6.7	Evaluation of Inter-Annotator Agreement using Chord Symbol Recall metrics across various distances: Binary (MIREX metrics), Tone-by-Tone, Mechanical, and Mechanical Consonance distances.	203
6.8	Overview of the chord recognition model architecture, comprising the preprocessing stage, conformer-based model, and symbolic chord decoder.	204
6.9	Comparison of the one-hot encoding of the root note with the root note encoding after applying consonance-based smoothing in the chord recognition model.	206
6.10	Example of chord label decoding for a $D:maj7/3$ chord as implemented in our chord recognition model.	207
6.11	Penultimate layer representation of notes C , $C\sharp$, and D from training without and with consonance-based smoothing in the chord recognition model.	209

List of Tables

3.1	Taxonomy of music ontologies based on their domain, scope, and the year of the latest release.	76
3.2	Overview of the modules in the Polifonia Ontology Network, including module name, prefix, description and repository name. . .	85
4.1	Overview of the 18 chord datasets currently included in ChoCo. . .	110
4.2	Links to key ChoCo resources: ontology, datasets, and knowledge graph.	113
4.3	Competency questions (CQs) addressed by the JAMS Ontology. .	120
4.4	Summary of the most common chord n-grams ($n = 2, 3, 4$), ranked by relative occurrence per annotation, with total n-gram occurrences in the ChoCo dataset.	131
4.5	Average coverage and accuracy of metadata and identifiers in the generated JAMS files, per ChoCo subset.	133
4.6	Evaluation of chord and key annotations in the generated JAMS files on test samples from the ChoCo dataset, averaged per subset and reported for times, durations, and labels.	134
4.7	Evaluation of chord conversions performed by music experts on a selection of ChoCo subsets.	136
4.8	Licensing information for each ChoCo subset, detailing the declared licence by the data curator.	142

List of Tables

5.1	Summary of the Kolmogorov-Smirnov tests used to detect statistically significant differences between groups/datasets on each metric in LHARP.	161
5.2	Performance of similarity algorithms on cover song detection in Harmory, highlighting the best-performing algorithms, with bold results indicating the top performance for the First Tier and Second Tier.	172
6.1	ChoCo partitions used for training and evaluating ChordSync. . .	198
6.2	Precision, Recall, and F1 Score for the HCDF method compared to the proposed ChordSync model.	199
6.3	Performance of ChordSync on the Schubert-Winterreise dataset compared to a standard DTW approach using the SyncToolbox library, along with performance metrics of the ChordSync method across all datasets, computed with alignment metrics from the mir_eval library.	200
6.4	Comparison of inter-annotator agreement evaluation metrics, presenting results using various metrics: Tone-by-Tone distance (TbT), Mechanical Distance (MD), and Mechanical Distance with Consonance (MDC).	204
6.5	ChoCo partitions used for training the Audio Chord Estimation model.	208
6.6	Performance comparison of the proposed Conformer architecture in the chord recognition model using different types of label smoothing, alongside evaluations against the bi-directional transformer architecture from Park et al.	209

List of Abbreviations

A

- ACE Audio Chord Estimation. 10–13, 15, 16, 18, 42, 181, 184, 186–189, 204, 210, 211, 219, 221, 222
- AI Artificial Intelligence. 57
- ASR Automatic Speech Recognition. 192, 195

B

- BRP Bag of Recurring Pattern. 156

C

- CNN Convolutional Neural Network. 152, 190, 192
- CQ Competency Question. 13, 16, 72, 78–82, 120, 216
- CQT Constant-Q Transform. 50, 195, 205
- CRF Conditional Random Field. 190
- CRNN Convolutional Recurrent Neural Network. 190
- CSAS Chord Sequence Alignment System. 8, 152

List of Abbreviations

CTC Connectionist Temporal Classification.
190, 193, 194, 197, 199, 211, 221, 230

D

DBN Dynamic Bayesian Network. 190

DCT Discrete Cosine Transform. 50

DL Deep Learning. 9, 53, 55, 182

DTW Dynamic Time Warping. 11, 15, 150, 168–
173, 189, 201, 210, 219, 221

F

FFT Fast Fourier Transform. 49, 50

G

GMN GUIDO Music Notation. 57, 58

GTTM Generative Theory of Tonal Music. 151

H

HCD Harmonic Change Detection. 199

HCQT Harmonic Constant-Q Transform. 50

HMM Hidden Markov Models. 189, 190

I

IE information entity. 90, 91

IR Information-Realisation. 90, 91

ISMIR International Society for Music Informa-
tion Retrieval. 39, 40

K

KE Knowledge Engineering. 216

KG Knowledge Graph. 15, 16, 59, 60, 70–73,
79, 87, 95, 96, 105–107, 119, 137, 143, 150,
151, 169, 216–218

KR Knowledge Representation. 13, 21, 22, 215

L

- LCS Longest Common Subsequence. 14, 150, 171
- LD Linked Data. 118, 119
- LS Label Smoothing. 191

M

- MEI Music Encoding Initiative. 55, 96
- MFCC Mel-Frequency Cepstral Coefficient. 50
- MIDI Musical Instrument Digital Interface. 40, 53, 54, 68, 189
- MIR Music Information Retrieval. 1–3, 5–7, 9–11, 13–15, 17, 21, 22, 33, 39–43, 49, 53, 56, 59, 61, 63–65, 68, 75, 97, 106–109, 111, 137–140, 143, 145, 181, 182, 185, 186, 188, 211, 215–218, 221
- ML Machine Learning. 55, 58, 182
- MMDL Multimodal Deep Learning. 61, 64
- MO Music Ontology. 69, 73, 74
- MRS Music Representation System. 52, 53

O

- ODP Ontology Design Pattern. 14, 16, 72, 79, 82, 83, 90, 91, 119
- OWL Web Ontology Language. 59

P

- PON Polifonia Ontology Network. 13, 16, 17, 72, 73, 79, 83, 84, 88, 90, 96, 97, 99, 100, 105, 106, 119, 124, 169, 170, 216, 217, 219, 220, 228

R

List of Abbreviations

RDF	Resource Description Framework. 59, 109
RL	Representation Learning. 50
RNN	Recurrent Neural Network. 190
RQ	Research Question. 13

S

SMF	Standard MIDI File. 54
SSM	Self Similarity Matrix. 166, 167
STFT	Short-Time Fourier Transform. 49, 50
SW	Semantic Web. 72, 97, 106–108, 137, 139, 140, 143, 154, 217

T

TPS	Tonal Pitch Space. 152, 165–167, 169, 170, 172–174, 179, 221, 230
TPSD	Tonal Pitch Step Distance. 8, 152, 171, 173

U

URI	Uniform Resource Identifier. 99, 119
-----	--------------------------------------

X

XCC	Explainable Computational Creativity. 153
XD	eXtreme Design. 13, 72, 79, 82, 99, 216
XML	Extensible Markup Language. 54, 55

Bibliography

- [1] Jakob Abeßer, Stefan Balke, Klaus Frieler, Martin Pfeiderer, and Meinard Müller. Deep learning for jazz walking bass transcription. In *Audio Engineering Society Conference: 2017 AES International Conference on Semantic Audio*. Audio Engineering Society, 2017.
- [2] Manel Achichi, Pasquale Lisena, Konstantin Todorov, Raphaël Troncy, and Jean Delahousse. DOREMUS: A graph of linked musical works. In *International Semantic Web Conference*, pages 3–19. Springer, 2018.
- [3] Elie Adam, E Noune, and Yasmina Yared. A system for music similarity search based on harmonic content. *Beirut, Lebanon*, 2010.
- [4] Alessandro Adamou, Simon Brown, Helen Barlow, Carlo Allocca, and Mathieu d’Aquin. Crowdsourcing Linked Data on listening experiences through reuse and enhancement of library data. *International Journal on Digital Libraries*, 20(1):61–79, 2019.
- [5] Eytan Agmon. Functional Harmony Revisited: A Prototype-Theoretic Approach. *Music Theory Spectrum*, 17(2):196–214, 10 1995.
- [6] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzett, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. MusicLM: Generating Music From Text, 2023.

- [7] Alo Allik, György Fazekas, and Mark B. Sandler. An Ontology for Audio Features. In Mandel et al. [254], pages 73–79.
- [8] Greg Aloupis, Thomas Fevens, Stefan Langerman, Tomomi Matsui, Antonio Mesa, Yurair Núñez Rodríguez, David Rappaport, and Godfried T. Toussaint. Algorithms for Computing Geometric Measures of Melodic Similarity. *Computer Music Journal*, 30(3):67–76, 2006.
- [9] William G Andrews and Molly Sclater. *Materials of Western Music*. Alfred Music Publishing, 1997.
- [10] Renzo Angles, Marcelo Arenas, Pablo Barceló, Aidan Hogan, Juan L. Reutter, and Domagoj Vrgoc. Foundations of Modern Query Languages for Graph Databases. *ACM Computing Surveys*, 50(5):68:1–68:40, 2017.
- [11] Renzo Angles and Claudio Gutiérrez. Survey of graph database models. *ACM Computing Surveys*, 40(1):1:1–1:39, 2008.
- [12] Marcelo Gabriel Armentano, Walter A. De Noni, and Hernán F. Cardoso. Genre classification of symbolic pieces of music. *J. Intell. Inf. Syst.*, 48(3):579–599, 2017.
- [13] MIDI Manufacturers Association. *The Complete MIDI 1.0 Detailed Specification: Incorporating All Recommended Practices*. MIDI Manufacturers Association, 1996.
- [14] Agnes Axelsson and Gabriel Skantze. Multimodal User Feedback During Adaptive Robot-Human Presentations. *Frontiers Comput. Sci.*, 3:741148, 2021.
- [15] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(2):423–443, 2019.
- [16] Ana M Barbancho, Isabel Barbancho, Lorenzo J Tardón, and Emilio Molina. *Database of Piano Chords: An Engineering View of Harmony*. Springer, 2013.
- [17] Mark A Bartsch and Gregory H Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE/ Transactions on Multimedia*, 7(1):96–104, 2005.

Bibliography

- [18] Sean Bechhofer, Simon Dixon, George Fazekas, Thomas Wilmering, and Kevin Page. Computational analysis of the live music archive. In Cumming et al. [77].
- [19] Sean Bechhofer, Kevin Page, and David De Roure. Hello Cleveland! Linked Data publication of live music archives. In *2013 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, pages 1–4. IEEE, 2013.
- [20] Juan Pablo Bello, Elaine Chew, and Douglas Turnbull, editors. *ISMIR 2008, 9th International Conference on Music Information Retrieval, ISMIR 2008*, September 2008.
- [21] Juan Pablo Bello and Jeremy Pickens. A Robust Mid-Level Representation for Harmonic Content in Music Signals. In ismir2005 [207], pages 304–311.
- [22] Emmanouil Benetos, Simon Dixon, Zhiyao Duan, and Sebastian Ewert. Automatic music transcription: An overview. *IEEE Signal Processing Magazine*, 36(1):20–30, 2019.
- [23] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives, 2013.
- [24] Ian D. Bent. musical notation, July 2024.
- [25] Ian D. Bent, David W. Hughes, Robert C. Provine, Richard Rastall, Anne Kilmer, David Hiley, Janka Szendrei, Thomas B. Payne, Margaret Bent, and Geoffrey Chew. Notation, 2001.
- [26] B. Benward and M.N. Saker. *Music in Theory and Practice*. Number v. 1 in Music in Theory and Practice. McGraw-Hill, 2003.
- [27] Jacopo Berardinis, Valentina Carriero, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The Music Meta Ontology: A Flexible Semantic Model for the Interoperability of Music Metadata. In Sarti et al. [349].
- [28] Adam Berenzweig, Beth Logan, Daniel P. W. Ellis, and Brian P. W. Whitman. A large-scale evaluation of acoustic and subjective music-similarity measures. *Comput. Music J.*, 28(2):63–76, June 2004.

- [29] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific american*, 284(5):34–43, may 2001.
- [30] Lawrence F. Bernstein and Joseph P. Olive. Computers and the 16th-century Chanson a pilot project at the University of Chicago. *Computers and the Humanities*, 3(3):153–160, Jan 1969.
- [31] Thierry Bertin-Mahieux, Douglas Eck, and Michael Mandel. Automatic tagging of audio: The state-of-the-art. In *Machine audition: Principles, Algorithms and Systems*, pages 334–352. IGI Global, 2011.
- [32] Emmanuel Bigand, Richard Parncutt, and Fred Lerdahl. Perception of musical tension in short chord sequences: The influence of harmonic function, sensory dissonance, horizontal motion, and musical training. *Perception & Psychophysics*, 58(1):125–141, Jan 1996.
- [33] Louis Bigo, Laurent Feisthauer, Mathieu Giraud, and Florence Levé. Relevance of musical features for cadence detection. In Gómez et al. [158], pages 355–361.
- [34] Rachel Bittner, Magdalena Fuentes, David Rubinstein, Andreas Jansson, Keunwoo Choi, and Thor Kell. mirdata: Software for Reproducible Usage of Datasets. In Flexer et al. [139], pages 99–106.
- [35] Eva Blomqvist, Karl Hammar, and Valentina Presutti. Engineering Ontologies with Patterns - The eXtreme Design Methodology. In *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, volume 25 of *Studies on the Semantic Web*. IOS Press, Amsterdam, 2016.
- [36] Eva Blomqvist, Valentina Presutti, Enrico Daga, and Aldo Gangemi. Experimenting with eXtreme Design. In *Knowledge Engineering and Management by the Masses. EKAW 2010*, volume 6317, pages 120–134. Springer, Berlin, Heidelberg, 2010.
- [37] Margaret A. Boden. Understanding creativity. *The Journal of Creative Behavior*, 1992.
- [38] Margaret A. Boden. *The creative mind: Myths and mechanisms*. Routledge, 2004.

Bibliography

- [39] Paul M. Bodily and Dan Ventura. Explainability: An Aesthetic for Aesthetics in Computational Creative Systems. In François Pachet, Anna Jordanous, and Carlos León, editors, *Proceedings of the Ninth International Conference on Computational Creativity, ICCC 2018, Salamanca, Spain, June 25-29, 2018*, pages 153–160. Association for Computational Creativity (ACC), 2018.
- [40] Carl Boettiger. *rdflib: A high level wrapper around the redland package for common rdf applications*, 2018.
- [41] Thomas Bottini, Valentina Anita Carriero, Jason Carvalho, Philippe Cathé, Fiorela Ciroku, Enrico Daga, Marilena Daquino, Achille Davy-Rigaux, Marco Guillotel-Nothmann, Gurrieri, Philo van Kemenade, Eleonora Marzi, Albert Meroño Peñuelala, Paul Mulholland, Elena Musumeci, Valentina Presutti, and Andrea Scharnhorst. D1.1 Roadmap and pilot requirements 1st version. Technical report, EU Commission, The Polifonia consortium, 2021.
- [42] Jean-Pierre Briot, Gaëtan Hadjeres, and François-David Pachet. *Deep learning techniques for music generation*, volume 1. Springer, 2020.
- [43] George Bruseker, Nicola Carboni, and Anaïs Guillem. Cultural heritage data management: the role of formal ontology and CIDOC CRM. *Heritage and archaeology in the digital age: acquisition, curation, and dissemination of spatial cultural heritage data*, pages 93–131, 2017.
- [44] Nick Bryan-Kinns, Berker Banar, Corey Ford, Courtney N. Reed, Yixiao Zhang, Simon Colton, and Jack Armitage. Exploring xai for the arts: Explaining latent space in generative music, 2023.
- [45] Michel Buffa, Elena Cabrio, Michael Fell, Fabien Gandon, Alain Giboin, Romain Hennequin, Franck Michel, Johan Pauwels, Guillaume Pellerin, Maroua Tikat, et al. The WASABI dataset: cultural, lyrics and audio analysis metadata about 2 million popular commercially released songs. In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*, pages 515–531. Springer, 2021.
- [46] Morgan Buisson, Pablo Alonso-Jiménez, and Dmitry Bogdanov. Ambiguity Modelling with Label Distribution Learning for Music Classification. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 611–615, 2022.

-
- [47] Benjamin Burger, Phillip M Maffettone, Vladimir V Gusev, Catherine M Aitchison, Yang Bai, Xiaoyan Wang, Xiaobo Li, Ben M Alston, Buyi Li, Rob Clowes, et al. A mobile robotic chemist. *Nature*, 583(7815):237–241, 2020.
- [48] John Ashley Burgoyne, Jonathan Wild, and Ichiro Fujinaga. An Expert Ground Truth Set for Audio Chord Recognition and Music Analysis. In Klapuri and Leider [219], pages 633–638.
- [49] Bryan R. Burnham, Emma Long, and Jake Zeide. Pitch direction on the perception of major and minor modes. *Attention, Perception, & Psychophysics*, 83(1):399–414, Jan 2021.
- [50] Alison Callahan, Jose Cruz-Toledo, Peter Ansell, and Michel Dumontier. Bio2RDF Release 2: Improved Coverage, Interoperability and Provenance of Life Science Linked Data. In Philipp Cimiano, Óscar Corcho, Valentina Presutti, Laura Hollink, and Sebastian Rudolph, editors, *The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings*, volume 7882 of *Lecture Notes in Computer Science*, pages 200–212. Springer, may 2013.
- [51] Julio José Carabias-Orti, Francisco J. Rodríguez-Serrano, Pedro Vera-Candeas, Nicolás Ruiz-Reyes, and Francisco J. Cañadas-Quesada. An audio to score alignment framework using spectral factorization and dynamic time warping. In Müller and Wiering [286], pages 742–748.
- [52] Valentina Anita Carriero, Fiorela Ciroku, Jacopo de Berardinis, Delfina Sol Martinez Pandiani, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. Semantic integration of mir datasets with the polifonia ontology network. In Lee et al. [239].
- [53] Valentina Anita Carriero et al. The landscape of ontology reuse approaches. *Applications and Practices in Ontology Design, Extraction, and Reasoning*, 49:21, 2020.
- [54] Valentina Anita Carriero, Aldo Gangemi, Maria Letizia Mancinelli, Ludovica Marinucci, Andrea Giovanni Nuzzolese, Valentina Presutti, and Chiara Veninata. ArCo: The Italian cultural heritage knowledge graph.

Bibliography

- In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II* 18, pages 36–52. Springer, 2019.
- [55] Valentina Anita Carriero, Aldo Gangemi, Maria Letizia Mancinelli, Andrea Giovanni Nuzzolese, Valentina Presutti, and Chiara Veninata. Pattern-based design applied to cultural heritage knowledge graphs. *Semantic Web*, 12(2):313–357, 2021.
- [56] Tristan Carsault, Jérôme Nika, and Philippe Esling. Using Musical Relationships Between Chord Labels in Automatic Chord Extraction Tasks. In Gómez et al. [158], pages 18–25.
- [57] Shan Carter and Michael Nielsen. Using artificial intelligence to augment human intelligence. *Distill*, 2(12):e9, 2017.
- [58] Yunkee Chae, Junghyun Koo, Sungho Lee, and Kyogu Lee. Exploiting time-frequency conformers for music audio enhancement. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM ’23, page 2362–2370, New York, NY, USA, 2023. Association for Computing Machinery.
- [59] Wei Chai and Barry Vercoe. Detection of key change in classical piano music. In ismir2005 [207], pages 468–473.
- [60] Ruofeng Chen and Ming Li. Music Structural Segmentation by Combining Harmonic and Timbral Information. In Klapuri and Leider [219], pages 477–482.
- [61] Tsung-Ping Chen, Li Su, et al. Functional Harmony Recognition of Symbolic Music Data with Multi-task Recurrent Neural Networks. In Gómez et al. [158], pages 90–97.
- [62] Samira Si-said Cherfi, Christophe Guillotel, Fayçal Hamdi, Philippe Rigaux, and Nicolas Travers. Ontology-Based Annotation of Music Scores. In Óscar Corcho, Krzysztof Janowicz, Giuseppe Rizzo, Ilaria Tiddi, and Daniel Garijo, editors, *Proceedings of the Knowledge Capture Conference, K-CAP 2017*, pages 10:1–10:4, New York, NY, USA, 2017. Association for Computing Machinery.

- [63] Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. Self-supervised learning with random-projection quantizer for speech recognition, 2022.
- [64] Pierre Choffé and Françoise Leresche. DOREMUS: connecting sources, enriching catalogues and user experience. In *24th IFLA World Library and Information Congress*, pages 1–20, 2016.
- [65] Keunwoo Choi, György Fazekas, Kyunghyun Cho, and Mark Sandler. A tutorial on deep learning for music information retrieval, 2018.
- [66] Anna-Maria Christodoulou, Olivier Lartillot, and Alexander Refsum Jensenius. Multimodal music datasets? Challenges and future goals in music processing. *International Journal of Multimedia Information Retrieval*, 13(3):37, Aug 2024.
- [67] Ching-Hua Chuan, Elaine Chew, et al. A hybrid system for automatic generation of style-specific accompaniment. In *Proceedings of the 4th international joint workshop on computational creativity*, pages 57–64. Goldsmiths, University of London London, 2007.
- [68] Kenneth Ward Church. Word2Vec. *Natural Language Engineering*, 23(1):155–162, 2017.
- [69] J.P. Clendinning and E.W. Marvin. *The Musician’s Guide to Theory and Analysis: Third Edition*. W.W. Norton, 2016.
- [70] Richard Cohn, Brian Hyer, Carl Dahlhaus, Julian Anderson, and Charles Wilson. *Harmony*, 2001.
- [71] Nick Collins, V Ruzicka, and Mick Grierson. Remixing AIs: mind swaps, hybridity, and splicing musical models. In *Proc. The Joint Conference on AI Music Creativity*, 2020.
- [72] Simon Colton, John William Charnley, and Alison Pease. Computational Creativity Theory: The FACE and IDEA Descriptive Models. In *ICCC*, pages 90–95. Mexico City, 2011.
- [73] Simon Colton and Dan Ventura. You Can’t Know my Mind: A Festival of Computational Creativity. In Simon Colton, Dan Ventura, Nada Lavrac, and Michael Cook, editors, *Proceedings of the Fifth International Conference*

Bibliography

- on Computational Creativity, ICCC 2014, Ljubljana, Slovenia, June 10-13, 2014*, pages 351–354. computationalcreativity.net, 2014.
- [74] Norman D Cook and Takashi X Fujisawa. The psychophysics of harmony perception: Harmony is a three-tone phenomenon. *Empirical Musicology Review*, 2006.
- [75] T. Crawford and L. Gibson. *Modern Methods for Musicology: Prospects, Proposals, and Realities*. Digital Research in the Arts and Humanities. Taylor & Francis, 2016.
- [76] Markus Cremer. A system for harmonic analysis of polyphonic music. In *Audio Engineering Society Conference: 25th International Conference: Metadata for Audio*. Audio Engineering Society, 2004.
- [77] Julie Cumming, Jin Ha Lee, Brian McFee, Markus Schedl, Johanna Devaney, Cory McKay, Eva Zangerle, and Timothy de Reuse, editors. *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020*, October 2020.
- [78] Sally Jo Cunningham, Zhiyao Duan, Xiao Hu, and Douglas Turnbull, editors. *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017*, October 2017.
- [79] Michael Scott Cuthbert and Christopher Ariza. music21 : A Toolkit for Computer-Aided Musicology and Symbolic Music Data. In Downie and Veltkamp [121], pages 637–642.
- [80] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. In Doina Precup and Yee Whye Teh, editors, *International conference on machine learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 894–903. PMLR, PMLR, 08 2017.
- [81] Enrico Daga, Luigi Asprino, Paul Mulholland, and Aldo Gangemi. Facade-X: An Opinionated Approach to SPARQL Anything. In Mehwish Alam, Paul Groth, Victor de Boer, Tassilo Pellegrini, and Harshvardhan J. Pandit, editors, *Volume 53: Further with Knowledge Graphs*, volume 53, pages 58–73. IOS Press, August 2021.

-
- [82] David Dalmazzo, Ken Déguernel, and Bob L. T. Sturm. The chordinator: Modeling music harmony by implementing transformer networks and token strategies. In *Artificial Intelligence in Music, Sound, Art and Design: 13th International Conference, EvoMUSART 2024, Held as Part of EvoStar 2024, Aberystwyth, UK, April 3–5, 2024, Proceedings*, page 52–66, Berlin, Heidelberg, 2024. Springer-Verlag.
- [83] Roger B. Dannenberg. A Brief Survey of Music Representation Issues, Techniques, and Systems. *Computer Music Journal*, 17, 11 1993.
- [84] Roger B. Dannenberg, William P. Birmingham, Bryan Pardo, Ning Hu, Colin Meek, and George Tzanetakis. A comparative evaluation of search techniques for query-by-humming using the MUSART testbed. *Journal of the American Society for Information Science and Technology*, 58(5):687–701, 2007.
- [85] Marilena Daquino and Enrico Daga. MusoW: Music data on the Web.
- [86] Marilena Daquino, Valentina Pasqual, and Francesca Tomasi. Knowledge Representation of digital Hermeneutics of archival and literary Sources. *Knowledge Representation of digital Hermeneutics of archival and literary Sources*, pages 59–76, 2020.
- [87] Marilena Daquino, Valentina Pasqual, Francesca Tomasi, and Fabio Vitali. Expressing Without Asserting in the Arts. In *CEUR WORKSHOP PROCEEDINGS*, volume 3160, 2022.
- [88] Jacopo de Berardinis, Valentina Anita Carriero, Nitisha Jain, Nicolas Lazari, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The polifonia ontology network: Building a semantic backbone for musical heritage. In Terry R. Payne, Valentina Presutti, Guilin Qi, María Poveda-Villalón, Giorgos Stoilos, Laura Hollink, Zoi Kaoudi, Gong Cheng, and Juanzi Li, editors, *The Semantic Web – ISWC 2023*, pages 302–322, Cham, 2023. Springer Nature Switzerland.
- [89] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The Music Annotation Pattern. In Vojtech Svátek, Valentina Anita Carriero, María Poveda-Villalón, Christian Kindermann, and Lu Zhou, editors, *Proceedings of the 13th Workshop on Ontology Design*

Bibliography

- and Patterns (WOP 2022) co-located with the 21th International Semantic Web Conference (ISWC 2022), Online, October 24, 2022*, volume 3352 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.
- [90] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. Choco: a chord corpus and a data transformation workflow for musical harmony knowledge graphs. *Scientific Data*, 10(1):641, 2023.
- [91] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The harmonic memory: a knowledge graph of harmonic patterns as a trustworthy framework for computational creativity. In Ying Ding, Jie Tang, Juan F. Sequeda, Lora Aroyo, Carlos Castillo, and Geert-Jan Houben, editors, *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, pages 3873–3882. ACM, 2023.
- [92] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The Harmonic Memory: a Knowledge Graph of harmonic patterns as a trustworthy framework for computational creativity. In *Proceedings of the ACM Web Conference 2023*, pages 3873–3882, 2023.
- [93] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. A Local Harmonic Agreement based on Recurring Patterns (LHARP). In *Paper under review*, 2024.
- [94] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. The Music Annotation Pattern. In Vojtěch Svátek, Valentina Anita Carriero, María Poveda, Christian Kindermann, and Lu Zhou, editors, *Proceedings of the 13th Workshop on Ontology Design and Patterns (WOP 2023)*, 2022.
- [95] Jacopo de Berardinis, Albert Meroño-Peñuela, Andrea Poltronieri, and Valentina Presutti. ChoCo: a Chord Corpus and a Data Transformation Workflow for Musical Harmony Knowledge Graphs. In *Manuscript under review*, 2023.
- [96] Jacopo de Berardinis, Michalis Vamvakaris, Angelo Cangelosi, and Eduardo Coutinho. Unveiling the hierarchical structure of music by multi-resolution

- community detection. *Transactions of the International Society for Music Information Retrieval*, 3(1):82–97, 2020.
- [97] Trevor de Clercq and David Temperley. A corpus analysis of rock harmony. *Popular Music*, 30(1):47–70, 2011.
- [98] Bas de Haas, José Pedro Magalhães, Dion ten Heggeler, Gijs Bekenkamp, and Tijmen Ruizendaal. Chordify: Chord transcription for the masses. In *Digital Humanities Conference Benelux*, June 2014.
- [99] W Bas De Haas, Matthias Robine, Pierre Hanna, Remco C Veltkamp, and Frans Wiering. Comparing approaches to the similarity of musical chord sequences. In *International Symposium on Computer Music Modeling and Retrieval*, pages 242–258. Springer, 2010.
- [100] W Bas De Haas, JP Rodrigues Magalhães, Remco C Veltkamp, Frans Wiering, et al. Harmtrace: Improving harmonic similarity estimation using functional harmony analysis. In Klapuri and Leider [219].
- [101] W. Bas de Haas, Martin Rohrmeier, Remco C. Veltkamp, and Frans Wiering. Modeling Harmonic Similarity Using a Generative Grammar of Tonal Harmony. In Hirata et al. [190], pages 549–554.
- [102] W Bas De Haas, Remco C Veltkamp, and Frans Wiering. Tonal Pitch Step Distance: a Similarity Measure for Chord Progressions. In Bello et al. [20], pages 51–56.
- [103] W. Bas de Haas, Remco C. Veltkamp, and Frans Wiering. TONAL PITCH STEP DISTANCE: A SIMILARITY MEASURE FOR CHORD PROGRESSIONS, September 2008.
- [104] W. Bas de Haas, Frans Wiering, and Remco C. Veltkamp. A geometrical distance measure for determining the similarity of musical harmony. *International Journal of Multimedia Information Retrieval*, 2(3):189–202, Sep 2013.
- [105] Blanca de Miguel-Molina and Rafael Boix-Doménech. Introduction: Music, from Intangible Cultural Heritage to the Music Industry. *Music as Intangible Cultural Heritage: Economic, Cultural and Social Identity*, pages 3–8, 2021.

Bibliography

- [106] Hélio de Oliveira and Raimundo Oliveira. Understanding midi: A painless tutorial on midi format, 05 2017.
- [107] Alceu de Souza Britto Jr., Fabien Gouyon, and Simon Dixon, editors. *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013*, November 2013.
- [108] Alessio Degani, Marco Dalai, Riccardo Leonardi, and Pierangelo Migliorati. Harmonic change detection for musical chords segmentation. In *2015 IEEE International Conference on Multimedia and Expo, ICME 2015, Turin, Italy, June 29 - July 3, 2015*, pages 1–6. IEEE Computer Society, 2015.
- [109] Yashar Deldjoo, Markus Schedl, and Peter Knees. Content-driven music recommendation: Evolution, state of the art, and challenges. *Computer Science Review*, 51:100618, 2024.
- [110] Jun-qi Deng and Yu-Kwong Kwok. Large Vocabulary Automatic Chord Estimation with an Even Chance Training Scheme. In Cunningham et al. [78], pages 531–536.
- [111] Johanna Devaney, Claire Arthur, Nathaniel Condit-Schultz, and Kirsten Nisula. Theme and variation encodings with roman numerals (TAVERN): A new data set for symbolic music analysis. In Müller and Wiering [286].
- [112] Deepika Devarajan. Happy Birthday Watson Discovery. IBM Cloud Blog, December 2017. <https://www.ibm.com/blogs/bluemix/2017/12/happy-birthday-watson-discovery/>.
- [113] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805, 2018.
- [114] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019*, volume 1, pages 4171–4186. Association for Computational Linguistics, 2019.

-
- [115] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A Generative Model for Music, 2020.
- [116] Bruno Di Giorgi, Massimiliano Zanoni, Augusto Sarti, and Stefano Tubaro. Automatic chord recognition based on the probabilistic modeling of diatonic modal harmony. In *Proceedings of the 8th International Workshop on Multidimensional Systems*, pages 1–6. VDE, 2013.
- [117] Claudia Diamantini, Domenico Potena, and Emanuele Storti. Ontology-driven KDD process composition. In *Advances in Intelligent Data Analysis VIII: 8th International Symposium on Intelligent Data Analysis, IDA 2009, Lyon, France, August 31-September 2, 2009. Proceedings 8*, pages 285–296. Springer, 2009.
- [118] Simon Dixon, David Bainbridge, and Rainer Typke, editors. *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007*. Austrian Computer Society, September 2007.
- [119] J Stephen Downie. Music information retrieval. *Annual review of information science and technology*, 37(1):295–340, 2003.
- [120] J. Stephen Downie. The Scientific Evaluation of Music Information Retrieval Systems: Foundations and Future. *Comput. Music. J.*, 28(2):12–23, 2004.
- [121] J. Stephen Downie and Remco C. Veltkamp, editors. *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010*. International Society for Music Information Retrieval, August 2010.
- [122] William Drabkin. Scale, 2001.
- [123] Eric Drott. Copyright, compensation, and commons in the music AI industry. *Creative Industries Journal*, 14(2):190–207, 2021.
- [124] Xingjian Du, Ke Chen, Zijie Wang, Bilei Zhu, and Zejun Ma. Bytecover2: Towards Dimensionality Reduction of Latent Embedding for Efficient Cover Song Identification. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 616–620, 2022.
- [125] M. du Sautoy. *The Creativity Code: How AI is learning to write, paint and think*. HarperCollins Publishers, 2019.

Bibliography

- [126] Wlodzislaw Duch. Intuition, insight, imagination and creativity. *IEEE Computational Intelligence Magazine*, 2(3):40–52, 2007.
- [127] Quang Tien Duong, Duc Huy Nguyen, Bao Thang Ta, Nhat Minh Le, and Van Hai Do. Improving self-supervised audio representation based on contrastive learning with conformer encoder. In *Proceedings of the 11th International Symposium on Information and Communication Technology, SoICT '22*, page 270–275, New York, NY, USA, 2022. Association for Computing Machinery.
- [128] Douglas Eck and Juergen Schmidhuber. A first look at music composition using LSTM recurrent neural networks. *Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale*, 103:48, 2002.
- [129] Vsevolod Eremenko, Emir Demirel, Baris Bozkurt, and Xavier Serra. JAAH: Audio-aligned jazz harmony dataset, June 2018.
- [130] Slim Essid and Gaël Richard. Fusion of Multimodal Information in Music Content Analysis. *Multimodal Music Processing*, 3:37–52, 2012.
- [131] Sebastian Ewert, Meinard Muller, and Peter Grosche. High resolution audio synchronization using chroma onset features. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1869–1872, 2009.
- [132] Riccardo Falco, Aldo Gangemi, Silvio Peroni, David Shotton, and Fabio Vitali. Modelling OWL ontologies with Graffoo. In *The Semantic Web: ESWC 2014 Satellite Events: ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers 11*, pages 320–325. Springer, 2014.
- [133] Ángel Faraldo, Emilia Gómez, Sergi Jordà, and Perfecto Herrera. Key estimation in electronic dance music. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20–23, 2016. Proceedings 38*, pages 335–347. Springer, 2016.
- [134] György Fazekas and Mark B. Sandler. The Studio Ontology Framework. In Klapuri and Leider [219].

-
- [135] György Fazekas, Yves Raimond, Kurt Jacobson, and Mark Sandler. An overview of Semantic Web activities in the OMRAS2 project. *Journal of New Music Research*, 39, December 2010.
- [136] Jonathan Feist. *Berklee Contemporary Music Notation*. Hal Leonard Corporation, 2017.
- [137] Mariano Fernandez-Lopez, Asuncion Gomez-Perez, and Natalia Juristo. Methontology: from ontological art towards ontological engineering. In *Proceedings of the AAAI97 Spring Symposium*, pages 33–40, Stanford, USA, March 1997.
- [138] Benjamin Fields, Kevin R. Page, David De Roure, and Tim Crawford. The segment ontology: Bridging music-generic and domain-specific. In *Proceedings of the 2011 IEEE International Conference on Multimedia and Expo, ICME 2011, 11-15 July, 2011, Barcelona, Catalonia, Spain*. IEEE Computer Society, 2011.
- [139] Arthur Flexer, Geoffroy Peeters, Julián Urbano, and Anja Volk, editors. *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019*, November 2019.
- [140] Luciano Floridi. Establishing the rules for building trustworthy AI. *Nature Machine Intelligence*, 1(6):261–262, 2019.
- [141] Jonathan Foote. Visualizing music and audio using self-similarity. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, pages 77–80, 1999.
- [142] Seth Forsgren and Hayk Martiros. Riffusion - Stable diffusion for real-time music generation, 2022.
- [143] Linton C Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.
- [144] Klaus Frieler and Daniel Müllensiefen. The simile algorithm for melodic similarity. In ismir2005 [207].
- [145] Hiromasa Fujihara, Masataka Goto, Jun Ogata, and Hiroshi G. Okuno. Lyricsynchronizer: Automatic synchronization system between musical audio signals and lyrics. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1252–1261, 2011.

Bibliography

- [146] Takuya Fujishima. Realtime Chord Recognition of Musical Sound: a System Using Common Lisp Music. In *Proceedings of the 1999 International Computer Music Conference, ICMC 1999, Beijing, China, October 22-27, 1999*. Michigan Publishing, 1999.
- [147] Aldo Gangemi. Ontology Design Patterns for Semantic Web Content. In Yolanda Gil, Enrico Motta, V. Richard Benjamins, and Mark A. Musen, editors, *The Semantic Web - ISWC 2005, 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005, Proceedings*, volume 3729 of *Lecture Notes in Computer Science*, pages 262–276. Springer, 2005.
- [148] Aldo Gangemi, Nicola Guarino, Claudio Masolo, Alessandro Oltramari, and Luc Schneider. Sweetening Ontologies with DOLCE. In Asunción Gómez-Pérez and V. Richard Benjamins, editors, *Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, 13th International Conference (EKAW 2002)*, pages 166–181, Berlin, Heidelberg, 2002. Springer, Springer Berlin Heidelberg.
- [149] Aldo Gangemi and Peter Mika. Understanding the Semantic Web through Descriptions and Situations. In Robert Meersman, Zahir Tari, and Douglas C. Schmidt, editors, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, pages 689–706, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
- [150] Aldo Gangemi and Silvio Peroni. The Information Realization Pattern. In Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnadhi, and Valentina Presutti, editors, *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, volume 25 of *Studies on the Semantic Web*, pages 299–312. IOS Press, 2016.
- [151] Aldo Gangemi and Valentina Presutti. Ontology Design Patterns. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, International Handbooks on Information Systems, pages 221–243. Springer, Berlin, Heidelberg, 2009.
- [152] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.

-
- [153] Shaghayegh Gharghabi, Yifei Ding, Chin-Chia Michael Yeh, Kaveh Kamgar, Liudmila Ulanova, and Eamonn Keogh. Matrix profile VIII: domain agnostic online semantic segmentation at superhuman performance levels. In *2017 IEEE international conference on data mining (ICDM)*, pages 117–126. IEEE, 2017.
 - [154] Deepanway Ghosal and Maheshkumar H Kolekar. Music genre recognition using deep neural networks and transfer learning. In *Interspeech*, pages 2087–2091, 2018.
 - [155] Konstantinos Giannos and Emilios Cambouropoulos. Symbolic Encoding of Simultaneities: Re-Designing the General Chord Type Representation. In *8th International Conference on Digital Libraries for Musicology, DLfM '21*, page 67–74, New York, NY, USA, 2021. Association for Computing Machinery.
 - [156] Stamatios Giannoulakis, Nicolas Tsapatsoulis, and Nikos Grammalidis. Metadata for intangible cultural heritage. In *Proceedings of the 13th international joint conference on computer vision, imaging and computer graphics theory and applications (VISAPP 2018)*, pages 634–645, 2018.
 - [157] Mathieu Giraud, Ken Déguernel, and Emilios Cambouropoulos. Fragmentations with Pitch, Rhythm and Parallelism Constraints for Variation Matching. In Mitsuko Aramaki, Olivier Derrien, Richard Kronland-Martinet, and Sølvi Ystad, editors, *Sound, Music, and Motion*, pages 298–312, Cham, 2014. Springer International Publishing.
 - [158] Emilia Gómez, Xiao Hu, Eric Humphrey, and Emmanouil Benetos, editors. *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018*, September 2018.
 - [159] Michael Good. MusicXML for Notation and Analysis. In *The Virtual Score, Volume 12: Representation, Retrieval, Restoration*. The MIT Press, 05 2001.
 - [160] Mark RH Gotham. Connecting the Dots: Engaging Wider Forms of Openness for the Mutual Benefit of Musicians and Musicologists. *Empirical Musicology Review*, 16(1):34–46, 2021.
 - [161] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka. RWC Music Database: Popular, Classical and Jazz Music Databases. In

Bibliography

- ISMIR 2002, 3rd International Symposium on Music Information Retrieval*, pages 287–288, October 2002.
- [162] Maarten Grachten, Josep Lluís Arcos, and Ramón López de Mántaras. Melodic similarity: Looking for a good abstraction level. In *ISMIR 2004, 5th International Conference on Music Information Retrieval*, October 2004.
- [163] Mark Granroth-Wilding and Mark Steedman. A robust parser-interpreter for jazz chord sequences. *Journal of New Music Research*, 43(4):355–374, 2014.
- [164] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning, ICML '06*, page 369–376, New York, NY, USA, 2006. Association for Computing Machinery.
- [165] George Grove, Stanley Sadie, and K. Marie Stolba. *The New Grove Dictionary of Music and Musicians*. Macmillan Publishers, London ; Washington, D.C., 1980.
- [166] Dan Gruen, Thyra Rauch, Sarah Redpath, and Stefan Ruettinger. The use of stories in user experience design. *International Journal of Human-Computer Interaction*, 14(3-4):503–534, 2002.
- [167] Michael Gruninger and Maria S. Fox. The role of competency questions in enterprise engineering. In *Benchmarking — Theory and Practice. IFIP Advances in Information and Communication Technology*, pages 83–95. Springer, Boston, MA, 1994.
- [168] Christophe Guillotel-Nothmann. Knowledge extraction and modelling in the project Thesaurus Musicarum Germanicarum. In *Prague DH Workshops Session III: Editions*, 2020.
- [169] Christophe Guillotel-Nothmann and Anne-Emmanuelle Ceulemans. Das diatonisch-chromatische System zur Zeit des Michael Praetorius. Eine digitale Neuerschließung des Syntagma Musicum (1619) in Verbindung mit dem Tanzzzyklus Terpsichore (1612). In *Musik im Umbruch. Michael Praetorius zum 400. Todestag*, Harrassowitz, 2021.

-
- [170] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer: Convolution-augmented transformer for speech recognition. In Helen Meng, Bo Xu, and Thomas Fang Zheng, editors, *Inter-speech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 5036–5040. ISCA, 2020.
- [171] Zixun Guo, Jaeyong Kang, and Dorien Herremans. A domain-knowledge-inspired music embedding space and a novel attention mechanism for symbolic music modeling. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence, AAAI’23/IAAI’23/EAAI’23*. AAAI Press, 2023.
- [172] Chitrallekha Gupta, Rong Tong, Haizhou Li, and Ye Wang. Semi-supervised lyrics and solo-singing alignment. In Gómez et al. [158].
- [173] Ferras Hamad, Isaac Liu, and Xian Xing Zhang. Food Discovery with Uber Eats: Building a Query Understanding Engine. Uber Engineering Blog, June 2018. <https://eng.uber.com/uber-eats-query-understanding/>.
- [174] Andrew Hankinson, Perry Roland, and Ichiro Fujinaga. The music encoding initiative as a document-encoding framework. In Klapuri and Leider [219].
- [175] Pierre Hanna, Matthias Robine, and Thomas Rocher. An alignment based system for chord sequence retrieval. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries*, pages 101–104, 2009.
- [176] Nicholas Harley and Geraint Wiggins. An ontology for abstract, hierarchical music representation. In Müller and Wiering [286].
- [177] J.P.E. Harper-Scott and J. Samson. *An Introduction to Music Studies*. Cambridge University Press, 2009.
- [178] Mickey Hart. Preserving Our Musical Heritage: A Musician’s Outreach to Audio Engineers. *Journal of the Audio Engineering Society*, 49(7/8):667–670, 2001.

Bibliography

- [179] Christopher Harte. Towards automatic extraction of harmony information from music signals, 2010.
- [180] Christopher Harte, Mark Sandler, and Martin Gasser. Detecting harmonic change in musical audio. In *Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia*, AMCMM '06, page 21–26, New York, NY, USA, 2006. Association for Computing Machinery.
- [181] Christopher Harte, Mark B. Sandler, Samer A. Abdallah, and Emilia Gómez. Symbolic Representation of Musical Chords: A Proposed Syntax for Text Annotations. In ismir2005 [207], pages 66–71.
- [182] Olaf Hartig. Foundations of RDF* and SPARQL* (An Alternative Approach to Statement-Level Metadata in RDF). In *Alberto Mendelzon Workshop on Foundations of Data Management*, 2017.
- [183] Curtis Hawthorne, Ian Simon, Adam Roberts, Neil Zeghidour, Josh Gardner, Ethan Manilow, and Jesse H. Engel. Multi-instrument Music Synthesis with Spectrogram Diffusion. In Rao et al. [330], pages 598–607.
- [184] Bruce Haynes and Peter Cooke. Pitch, 2001.
- [185] Qi He, Bee-Chung Chen, and Deepak Agarwal. Building The LinkedIn Knowledge Graph. LinkedIn Blog, October 2016. <https://engineering.linkedin.com/blog/2016/10/building-the-linkedin-knowledge-graph>.
- [186] James A. Hendler, Jeanne Holm, Chris Musialek, and George Thomas. US Government Linked Open Data: Semantic.data.gov. *IEEE Intelligent Systems*, 27(3):25–31, 2012.
- [187] Johannes Hentschel, Fabian C. Moss, Andrew McLeod, Markus Neuwirth, and Martin Rohrmeier. Towards a Unified Model of Chords in Western Harmony. In Stefan Münnich and David Rizo, editors, *Music Encoding Conference Proceedings 2021*, pages 143–149. Humanities Commons, 2022.
- [188] Johannes Hentschel, Markus Neuwirth, and Martin Rohrmeier. The annotated Mozart Sonatas: Score, Harmony, and Cadence. *Transactions of the International Society for Music Information Retrieval*, 4(1), 2021.

-
- [189] Paul Hindemith and Arthur Mendel. *The Craft of Musical Composition. 1. Theoretical Part*. Schott, 1970.
- [190] Keiji Hirata, George Tzanetakis, and Kazuyoshi Yoshii, editors. *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009*. International Society for Music Information Retrieval, October 2009.
- [191] Thomas S. Hischak and Dai Griffiths. *Lyrics*, 2001.
- [192] Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnadhi, and Valentina Presutti, editors. *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, volume 25 of *Studies on the Semantic Web*. IOS Press, Amsterdam, 2016.
- [193] Pascal Hitzler, Markus Krötzsch, Bijan Parsia, Peter F. Patel-Schneider, and Sebastian Rudolph. OWL 2 Web Ontology Language Primer (Second Edition), W3C Recommendation 11 December 2012. W3c recommendation, World Wide Web Consortium, dec 2012.
- [194] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d’Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. Knowledge graphs. *Synthesis Lectures on Data, Semantics, and Knowledge*, 12(2):1–257, 2021.
- [195] Holger Hoos, Keith A. Hamel, Kai Renz, and Jürgen Kilian. Representing score-level music using the GUIDO music-notation format. *Computing in Musicology*, 12, 2001.
- [196] Holger H Hoos, Keith Hamel, Kai Renz, and Jürgen Kilian. The guido notation format: A novel approach for adequately representing score-level music. In *ICMC*, volume 98, pages 451–454, 1998.
- [197] Cheng-Zhi Anna Huang, Tim Cooijmans, Adam Roberts, Aaron C. Courville, and Douglas Eck. Counterpoint by Convolution. In Flexer et al. [139], pages 211–218.
- [198] Cheng-Zhi Anna Huang, Hendrik Vincent Koops, Ed Newton-Rex, Monica Dinculescu, and Carrie J Cai. AI song contest: Human-AI co-creation in songwriting. In Cumming et al. [77].

Bibliography

- [199] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Ian Simon, Curtis Hawthorne, Noam Shazeer, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck. Music Transformer: Generating Music with Long-Term Structure. In *International Conference on Learning Representations*, 2019.
- [200] Jiawen Huang, Emmanouil Benetos, and Sebastian Ewert. Improving lyrics alignment through joint pitch detection, 2022.
- [201] Yin-Fu Huang, Sheng-Min Lin, Huan-Yu Wu, and Yu-Siou Li. Music genre classification based on local feature selection using a self-adaptive harmony search algorithm. *Data Knowl. Eng.*, 92:60–76, 2014.
- [202] Eric J. Humphrey, Taemin Cho, and Juan Pablo Bello. Learning a robust Tonnetz-space transform for automatic chord recognition. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2012, Kyoto, Japan, March 25-30, 2012*, pages 453–456. IEEE, 2012.
- [203] Eric J. Humphrey, Justin Salamon, Oriol Nieto, Jon Forsyth, Rachel M. Bittner, and Juan P. Bello. JAMS: A JSON annotated music specification for reproducible MIR research. In Wang et al. [399], pages 591–596.
- [204] David Huron. Music information processing using the humdrum toolkit: Concepts, examples, and lessons. *Computer Music Journal*, 26(2):11–26, 2002.
- [205] Brian Hyer. *Tonality*, 2001.
- [206] Antoine Isaac and Bernhard Haslhofer. Europeana linked open data–data. europeana. eu. *Semantic Web*, 4(3):291–297, 2013.
- [207] *ISMIR 2005, 6th International Conference on Music Information Retrieval*, September 2005.
- [208] Kurt Jacobson, Yves Raimond, and Mark B. Sandler. An Ecosystem for Transparent Music Similarity in an Open World. In Hirata et al. [190], pages 33–38.
- [209] Alexander Refsum Jensenius. Best versus good enough practices for open music research. *Empirical Musicology Review*, 16(1):5–15, 2021.

-
- [210] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):494–514, 2022.
- [211] Junyan Jiang, Ke Chen, Wei Li, and Gus Xia. Large-vocabulary Chord Transcription Via Chord Structure Decomposition. In Flexer et al. [139], pages 644–651.
- [212] Cyril Joder, Slim Essid, and Gaël Richard. Learning optimal features for polyphonic audio-to-score alignment. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 21(10):2118–2128, 2013.
- [213] Jim Jones, Diego de Siqueira Braga, Kleber Tertuliano, and Tomi Kauppinen. MusicOWL: The Music Score Ontology. In *Proceedings of the International Conference on Web Intelligence*, WI '17, pages 1222–1229, New York, NY, USA, 2017. Association for Computing Machinery.
- [214] Spyridon Kantarelis, Edmund Dervakos, Natalia Kotsani, and Giorgos Stamou. Functional Harmony Ontology: Musical Harmony Analysis with Description Logics. *Web Semant.*, 75(C), January 2023.
- [215] Emmanouil Karystinaios and Gerhard Widmer. Cadence detection in symbolic classical music using graph neural networks. In Rao et al. [330].
- [216] Hyon Hee Kim. A Semantically Enhanced Tag-Based Music Recommendation Using Emotion Ontology. In *Intelligent Information and Database Systems*, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [217] Jaehun Kim, Julián Urbano, Cynthia Liem, and Alan Hanjalic. One Deep Music Representation to Rule Them All?: A Comparative Analysis of Different Representation Learning Strategies. *Neural Computing and Applications*, 32(4):1067–1093, 2020.
- [218] Arto Klami, Theo Damoulas, Ola Engkvist, Patrick Rinke, and Samuel Kaski. Virtual laboratories: transforming research with ai. *Data-Centric Engineering*, 5:e19, 2024.
- [219] Anssi Klapuri and Colby Leider, editors. *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011*. University of Miami, October 2011.

Bibliography

- [220] Peter Knees, Ángel Faraldo Pérez, Herrera Boyer, Richard Vogl, Sebastian Böck, Florian Hörschläger, Mickael Le Goff, et al. Two data sets for tempo estimation and key detection in electronic dance music annotated from user corrections. In Müller and Wiering [286], pages 364–370.
- [221] Peter Knees and Markus Schedl. Music similarity and retrieval. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, page 1125, New York, NY, USA, 2013. Association for Computing Machinery.
- [222] Peter Knees and Markus Schedl. A survey of music similarity and recommendation from music context data. *ACM Trans. Multimedia Comput. Commun. Appl.*, 10(1), dec 2013.
- [223] Stefan Koelsch. Toward a neural basis of music perception—a review and updated model. *Frontier in Psychology*, 2:110, 2011.
- [224] Sefki Kolozali, Mathieu Barthet, György Fazekas, and Mark B Sandler. Knowledge Representation Issues in Musical Instrument Ontology Design. In Klapuri and Leider [219], pages 465–470.
- [225] Hendrik Vincent Koops, Bas de Haas, John Ashley Burgoyne, Jeroen Bransen, Anna Kent-Muller, and Anja Volk. Annotator subjectivity in harmony annotations of popular music. *Journal of New Music Research*, 48(3):232–252, 2019.
- [226] Hendrik Vincent Koops, W. Bas de Haas, Jeroen Bransen, and Anja Volk. Automatic chord label personalization through deep learning of shared harmonic interval profiles. *Neural Comput. Appl.*, 32(4):929–939, February 2020.
- [227] Hendrik Vincent Koops, W. Bas de Haas, John Ashley Burgoyne, Jeroen Bransen, Anna Kent-Muller, and Anja Volk. Annotator subjectivity in harmony annotations of popular music. *Journal of New Music Research*, 48(3):232–252, may 2019.
- [228] Filip Korzeniowski and Gerhard Widmer. A fully convolutional deep auditory model for musical chord recognition. In Francesco A. N. Palmieri, Aurelio Uncini, Kostas I. Diamantaras, and Jan Larsen, editors, *26th IEEE International Workshop on Machine Learning for Signal Processing, MLSP*

- 2016, Vietri sul Mare, Salerno, Italy, September 13-16, 2016, pages 1–6. IEEE, 2016.
- [229] Harald Kosch. MPEG-7 and multimedia database systems. *SIGMOD Rec.*, 31(2):34–39, June 2002.
- [230] S.M. Kostka and D. Payne. *Tonal Harmony, with an Introduction to Twentieth-century Music*. McGraw-Hill, 2004.
- [231] Dominik Kowald, Markus Schedl, and Elisabeth Lex. The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, pages 35–42, Cham, 2020. Springer International Publishing.
- [232] Michael Krause, Christof Weiß, and Meinard Müller. Soft dynamic time warping for multi-pitch estimation and beyond. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.
- [233] Arun Krishnan. Making search easier: How Amazon’s Product Graph is helping customers find products more easily. Amazon Blog, August 2018. <https://blog.aboutamazon.com/innovation/making-search-easier>.
- [234] Allison Lahnala, Gauri Kambhatla, Jiajun Peng, Matthew Whitehead, Gillian Minnehan, Eric Guldán, Jonathan K. Kummerfeld, Anıl Çamcı, and Rada Mihalcea. Chord Embeddings: Analyzing What They Capture and Their Role for Next Chord Prediction and Artist Attribute Prediction. In Juan Romero, Tiago Martins, and Nereida Rodríguez-Fernández, editors, *Artificial Intelligence in Music, Sound, Art and Design*, volume 12693 of *Lecture Notes in Computer Science*, pages 171–186, Cham, 2021. Springer International Publishing.
- [235] S.G. Laitz. *The Complete Musician: An Integrated Approach to Tonal Theory, Analysis, and Listening*. Number v. 1 in The Complete Musician: An Integrated Approach to Tonal Theory, Analysis, and Listening. Oxford University Press, 2008.
- [236] Agnieszka Lawrynowicz. Creative AI: A new avenue for the Semantic Web? *Semantic Web*, 11:69–78, 2020.

Bibliography

- [237] Nicolas Lazzari, Andrea Poltronieri, and Valentina Presutti. Pitchclass2vec: Symbolic Music Structure Segmentation with Chord Embeddings. In Allegra De Filippo, Michela Milano, Valentina Presutti, and Alessandro Saffiotti, editors, *Proceedings of the 1st Workshop on Artificial Intelligence and Creativity co-located with 21th International Conference of the Italian Association for Artificial Intelligence(AIxA 2022), Udine, Italy, November 28 - December 3, 2022*, volume 3278 of *CEUR Workshop Proceedings*, pages 14–30. CEUR-WS.org, 2022.
- [238] Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. Prov-o: The prov ontology, 2013.
- [239] Jin Ha Lee, Alexander Lerch, Zhiyao Duan, Juhan Nam, Preeti Rao, Peter van Kranenburg, and Ajay Srinivasamurthy, editors. *Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR 2021*, November 2021.
- [240] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.
- [241] A. Lerch. *An Introduction to Audio Content Analysis: Music Information Retrieval Tasks and Applications*. Wiley, 2022.
- [242] Fred Lerdahl. Tonal Pitch Space. *Music Perception: An Interdisciplinary Journal*, 5(3):315–349, 1988.
- [243] Fred Lerdahl and Ray Jackendoff. *A generative theory of tonal music*. The MIT Press, Cambridge. MA, 1983.
- [244] Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xingran Chen, Hanzhi Yin, Chenghua Lin, Anton Ragni, Emmanouil Benetos, Norbert Gyenge, Roger Dannenberg, Ruibo Liu, Wenhua Chen, Gus Xia, Yemin Shi, Wenhao Huang, Yike Guo, and Jie Fu. MERT: Acoustic Music Understanding Model with Large-Scale Self-supervised Training. In *The Twelfth International Conference on Learning Representations*, 2024.

-
- [245] Julian Lienen and Eyke Hüllermeier. From Label Smoothing to Label Relaxation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8583–8591, May 2021.
- [246] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society, 2017.
- [247] Mark Lindley, Murray Campbell, and Clive Greated. Interval, 2001.
- [248] Pasquale Lisena, Konstantin Todorov, Cécile Cecconi, Françoise Leresche, Isabelle Canno, Frédéric Puyrenier, Martine Voisin, Thierry Le Meur, and Raphaël Troncy. Controlled vocabularies for music metadata. In Gómez et al. [158], pages 424–431.
- [249] Pasquale Lisena and Raphaël Troncy. DOing REusable MUSical Data (DOREMUS). In *Proceedings of Workshops and Tutorials of the 9th International Conference on Knowledge Capture (K-CAP2017), Austin, Texas, USA, December 4th, 2017*, volume 2065 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2017.
- [250] Chihuang Liu and Joseph JaJa. Class-Similarity Based Label Smoothing for Confidence Calibration. In Igor Farkas, Paolo Masulli, Sebastian Otte, and Stefan Wermter, editors, *Artificial Neural Networks and Machine Learning – ICANN 2021*, pages 190–201, Cham, 2021. Springer International Publishing.
- [251] Maria Teresa Llano, Mark d’Inverno, Matthew Yee-King, Jon McCormack, Alon Ilisar, Alison Pease, and Simon Colton. Explainable Computational Creativity. In F. Amílcar Cardoso, Penousal Machado, Tony Veale, and João Miguel Cunha, editors, *Proceedings of the Eleventh International Conference on Computational Creativity, ICC3 2020, Coimbra, Portugal, September 7-11, 2020*, pages 334–341. Association for Computational Creativity (ACC), 2020.
- [252] Yinghao Ma, Anders Øland, Anton Ragni, Bleiz MacSen Del Sette, Charalampos Saitis, Chris Donahue, Chenghua Lin, Christos Plachouras, Emmanouil Benetos, Elona Shatri, Fabio Morreale, Ge Zhang, György Fazekas,

Bibliography

- Gus Xia, Huan Zhang, Ilaria Manco, Jiawen Huang, Julien Guinot, Liwei Lin, Luca Marinelli, Max W. Y. Lam, Megha Sharma, Qiuqiang Kong, Roger B. Dannenberg, Ruibin Yuan, Shangda Wu, Shih-Lun Wu, Shuqi Dai, Shun Lei, Shiyin Kang, Simon Dixon, Wenhui Chen, Wenhao Huang, Xingjian Du, Xingwei Qu, Xu Tan, Yizhi Li, Zeyue Tian, Zhiyong Wu, Zhizheng Wu, Ziyang Ma, and Ziyu Wang. Foundation Models for Music: A Survey, 2024.
- [253] Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood, and Imran Makhdoom. Automatic speech recognition: a survey. *Multimedia Tools and Applications*, 80:9411–9457, 2021.
- [254] Michael I. Mandel, Johanna Devaney, Douglas Turnbull, and George Tzanetakis, editors. *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016*, August 2016.
- [255] Alan Marsden. Generative structural representation of tonal music. *Journal of New Music Research*, 34(4):409–428, 2005.
- [256] Alan Marsden. Music Analysis by Computer: Ontology and Epistemology. In David Meredith, editor, *Computational Music Analysis*, pages 3–28. Springer, 2016.
- [257] Ninon Lizé Masclef, Andrea Vaglio, and Manuel Moussallam. User-centered evaluation of lyrics-to-audio alignment. In Lee et al. [239], pages 420–427.
- [258] Matthias Mauch, Chris Cannam, Matthew Davies, Simon Dixon, Christopher Harte, Sefki Kolozali, Dan Tidhar, and Mark Sandler. OMRAS2 meta-data project 2009. In Hirata et al. [190].
- [259] Matthias Mauch and Simon Dixon. Approximate Note Transcription for the Improved Identification of Difficult Chords. In Downie and Veltkamp [121], pages 135–140.
- [260] Matthias Mauch, Simon Dixon, Christopher Harte, Michael A. Casey, and Benjamin Fields. Discovering Chord Idioms Through Beatles and Real Book Songs. In Dixon et al. [118], pages 255–258.
- [261] Matthias Mauch, Hiromasa Fujihara, and Masataka Goto. Lyrics-to-audio alignment and phrase-level segmentation using incomplete internet-

- style chord annotations. In *7th Sound and Music Computing Conference (SMC2010)*, 01 2010.
- [262] Brian McFee, Luke Barrington, and Gert Lanckriet. Learning Content Similarity for Music Recommendation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 20(8):2207–2218, 2012.
- [263] Brian McFee and Juan Pablo Bello. Structured training for large-vocabulary chord recognition. In Cunningham et al. [78], pages 188–194.
- [264] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.*, 3(29):861, 2018.
- [265] Neil McLachlan, David Marco, Maria Light, and Sarah Wilson. Consonance and pitch. *Journal of Experimental Psychology: General*, 142(4):1142, 2013.
- [266] Andrew Mcleod, Xavier Suermondt, Yannis Rammos, Steffen Herff, and Martin A. Rohrmeier. Three Metrics for Musical Chord Label Evaluation. In *Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '22*, page 47–53, New York, NY, USA, 2023. Association for Computing Machinery.
- [267] Matt McVicar, Raúl Santos-Rodríguez, Yizhao Ni, and Tijl De Bie. Automatic Chord Estimation from Audio: A Review of the State of the Art. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(2):556–575, 2014.
- [268] Rick Meerwaldt, Albert Meroño-Peñuela, and Stefan Schlobach. Mixing Music as Linked Data: SPARQL-based MIDI Mashups. In *WHiSe@ ISWC*, pages 87–98, 2017.
- [269] Antonello Meloni, Diego Reforgiato Recupero, and Aldo Gangemi. AMR2FRED, A Tool for Translating Abstract Meaning Representation to Motif-Based Linguistic Knowledge Graphs. In Eva Blomqvist, Katja Hose, Heiko Paulheim, Agnieszka Ławrynowicz, Fabio Ciravegna, and Olaf Hartig, editors, *The Semantic Web: ESWC 2017 Satellite Events*, pages 43–47, Portorož, Slovenia, 2017. Springer International Publishing.
- [270] Judith Merges, Martin Scholz, and Guenther Goerz. Erlangen Implementation of FRBRoo. In *CIDOC 2012*, 2012.

Bibliography

- [271] Albert Meroño-Peñuela, Jacopo de Berardinis, Valentina Anita Carriero, Mari Wigham, Andrea Poltronieri, Fiorela Ciroku, Christophe Guillotel-Nothmann, and Philippe Rigaux. D2.1: Ontology-based knowledge graphs for music objects (V1.0), 2021.
- [272] Albert Meroño-Peñuela, Rinke Hoekstra, Aldo Gangemi, Peter Bloem, Reinier de Valk, Bas Stringer, Berit Janssen, Victor de Boer, Alo Allik, Stefan Schlobach, and Kevin Page. The MIDI Linked Data Cloud. In *The Semantic Web – ISWC 2017*, pages 156–164, Cham, 2017. Springer, Springer International Publishing.
- [273] Gianluca Micchi, Mark Gotham, and Mathieu Giraud. Not all roads lead to Rome: Pitch representation and model architecture for automatic harmonic analysis. *Transactions of the International Society for Music Information Retrieval*, 3(1):42–54, 2020.
- [274] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory F. Damos, Erich Elsen, David García, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed Precision Training. *CoRR*, abs/1710.03740, 2017.
- [275] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.
- [276] Steven Mithen. *Creativity in human evolution and prehistory*. Routledge, 2005.
- [277] Efthymia Moraitou, Yannis Christodoulou, and George Caridakis. Semantic models and services for conservation and restoration of cultural heritage: A comprehensive survey. *Semantic Web*, 14(2):261–291, 2023.
- [278] Fabio Morreale. Where Does the Buck Stop? Ethical and Political Issues with AI in Music Creation. *Transactions of the International Society for Music Information Retrieval*, Jul 2021.
- [279] Alia Morsi and Xavier Serra. Bottlenecks and solutions for audio to score alignment research. In Rao et al. [330], pages 272–279.

-
- [280] Lindasalwa Muda, Mumtaj Begam, and I. Elamvazuthi. Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques. *CoRR*, 2010.
- [281] Daniel Müllensiefen and Marc Pendzich. Court decisions on music plagiarism and the predictive value of similarity algorithms. *Musicae Scientiae*, 13(1_suppl):257–295, 2009.
- [282] Meinard Müller. *Fundamentals of music processing: Using Python and Jupyter notebooks*, volume 2. Springer, 2021.
- [283] Meinard Müller, Andreas Arzt, Stefan Balke, Matthias Dorfer, and Gerhard Widmer. Cross-Modal Music Retrieval and Applications: An Overview of Key Methodologies. *IEEE Signal Process. Mag.*, 36(1):52–62, 2019.
- [284] Meinard Müller and Anssi Klapuri. Music signal processing. In *Academic Press Library in Signal Processing*, volume 4, pages 713–756. Elsevier, 2014.
- [285] Meinard Müller, Henning Mattes, and Frank Kurth. An efficient multi-scale approach to audio synchronization. In *ISMIR 2006, 7th International Conference on Music Information Retrieval*, pages 192–197, October 2006.
- [286] Meinard Müller and Frans Wiering, editors. *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015*, October 2015.
- [287] Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. When Does Label Smoothing Help? *CoRR*, abs/1906.02629, 2019.
- [288] Meinard Müller, Yigitcan Özer, Michael Krause, Thomas Prätzlich, and Jonathan Driedger. Sync toolbox: A python package for efficient, robust, and accurate music synchronization. *Journal of Open Source Software*, 6(64):3434, 2021.
- [289] Yizhao Ni, Matt McVicar, Raul Santos-Rodriguez, and Tijl De Bie. An end-to-end machine learning system for harmonic analysis of music. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 20(6):1771–1783, 2012.
- [290] Han-Wen Nienhuys and Jan Nieuwenhuizen. Lilypond, a system for automated music engraving. In *Proceedings of the xiv colloquium on musical informatics (xiv cim 2003)*, volume 1, pages 167–171. Citeseer, 2003.

Bibliography

- [291] Oriol Nieto and Juan Pablo Bello. Systematic Exploration of Computational Music Structure Research. In Mandel et al. [254], pages 547–553.
- [292] Nottingham Database. <https://ifdo.ca/~seymour/nottingham/nottingham.html>.
- [293] Natasha F. Noy, Yuqing Gao, Anshu Jain, Anant Narayanan, Alan Patterson, and Jamie Taylor. Industry-scale Knowledge Graphs: Lessons and Challenges. *ACM Queue*, 17(2):20, 2019.
- [294] Nicola Orio and Antonio Rodà. A Measure of Melodic Similarity based on a Graph Representation of the Music Structure. In Hirata et al. [190], pages 543–548.
- [295] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS ’22, Red Hook, NY, USA, 2024. Curran Associates Inc.
- [296] Francois Pachet. Knowledge management and musical metadata. *Idea Group*, 12, 2005.
- [297] Elias Pampalk. *Computational models of music similarity and their application in music information retrieval*. PhD thesis, Technische Universität Wien, 2006.
- [298] Letitia Parcalabescu, Nils Trost, and Anette Frank. What is Multimodality? In Lucia Donatelli, Nikhil Krishnaswamy, Kenneth Lai, and James Pustejovsky, editors, *Proceedings of the 1st Workshop on Multimodal Semantic Representations (MMSR)*, pages 1–10, Groningen, Netherlands (Online), June 2021. Association for Computational Linguistics.
- [299] Ferran Parés, Dario Garcia Gasulla, Armand Vilalta, Jonatan Moreno, Eduard Ayguadé, Jesús Labarta, Ulises Cortés, and Toyotaro Suzumura. Fluid communities: A competitive, scalable and diverse community detection al-

- gorithm. In *International Conference on Complex Networks and their Applications*, pages 229–240. Springer, 2017.
- [300] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*. ISCA, September 2019.
- [301] Jonggwon Park, Kyoyun Choi, Sungwook Jeon, Dokyun Kim, and Jonghun Park. A Bi-Directional Transformer for Musical Chord Recognition. In Flexer et al. [139], pages 620–627.
- [302] Johan Pauwels, Ken O’Hanlon, Emilia Gómez, and Mark B. Sandler. 20 years of automatic chord recognition from audio. In Flexer et al. [139], pages 54–63.
- [303] D. Pedler. *The Songwriting Secrets Of The Beatles*. Music Sales, 2010.
- [304] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global Vectors for Word Representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL, 2014.
- [305] Carlos Pérez-Sancho, David Rizo, José M Iñesta, Pedro J Ponce De León, Stefan Kersten, and Rafael Ramirez. Genre classification of music by tonal harmony. *Intelligent Data Analysis*, 14(5):533–545, 2010.
- [306] Silvio Peroni. A simplified agile methodology for ontology development. In *OWL: Experiences and Directions—Reasoner Evaluation*, pages 55–69. Springer, 2016.
- [307] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

Bibliography

- [308] Bastian Pfleging, Stefan Schneegeß, and Albrecht Schmidt. Multimodal interaction in the car: combining speech and gestures on the steering wheel. In *AutomotiveUI*, pages 155–162. ACM, 2012.
- [309] Martin Pfeleiderer, Klaus Frieler, Jakob Abeßer, Wolf-Georg Zaddach, and Benjamin Burkhart, editors. *Inside the Jazzomat - New Perspectives for Jazz Research*. Schott Campus, 2017.
- [310] Azzurra Pini, Jer Hayes, Connor Upton, and Medb Corcoran. AI Inspired Recipes: Designing Computationally Creative Food Combos. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019. Association for Computing Machinery.
- [311] Helena Sofia Pinto, Steffen Staab, and Christoph Tempich. DILIGENT: Towards a fine-grained methodology for DIstributed, Loosely-controlled and evolvInG Engineering of oNTologies. In *ECAI*, volume 16, page 393. Cite-seer, 2004.
- [312] W. Piston. *Harmony*. W. W. Norton, 1962.
- [313] C.J. Plack, A.J. Oxenham, and R.R. Fay. *Pitch: Neural Coding and Perception*. Online access: LexisNexis Nexis Advance UK. Springer, 2005.
- [314] Andrea Poltronieri and Aldo Gangemi. The HaMSE ontology: Using semantic technologies to support music representation interoperability and musicological analysis. In *Proceedings of the 1st workshop on multisensory data and knowledge (MDK)*, 2021.
- [315] Andrea Poltronieri and Aldo Gangemi. The Music Note Ontology. In Karl Hammar, Cogan Shimizu, Hande Küçük McGinty, Luigi Asprino, and Valentina Anita Carriero, editors, *Proceedings of the 12th Workshop on Ontology Design and Patterns (WOP 2021), Online, October 24, 2021.*, November 2021.
- [316] Andrea Poltronieri, Valentina Presutti, and Martín Rocamora. ChordSync: A Conformer-based Audio-to-Chord Synchroniser. In *Proceedings of the 2024 Sound and Music Computing Conference*. Sound and Music Computing Network, July 2024.

-
- [317] Andrea Poltronieri, Valentina Presutti, and Martín Rocamora. From Discord to Consonance: Consonance-based Label Smoothing for Improved Audio Chord Estimation. In *Paper under review*, 2024.
- [318] Valentina Presutti, Enrico Daga, Aldo Gangemi, and Eva Blomqvist. eXtreme Design with Content Ontology Design Patterns. In Eva Blomqvist, Kurt Sandkuhl, François Scharffe, and Vojtech Svátek, editors, *Proceedings of the Workshop on Ontology Patterns (WOP 2009)*, volume 516 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.
- [319] Thomas Prätzlich, Jonathan Driedger, and Meinard Müller. Memory-restricted multiscale dynamic time warping. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 569–573, 2016.
- [320] Michael Pulis and Josef Bajada. Siamese neural networks for content-based cold-start music recommendation. In *Proceedings of the 15th ACM Conference on Recommender Systems, RecSys '21*, page 719–723, New York, NY, USA, 2021. Association for Computing Machinery.
- [321] Alexander M Putman and Robert M Keller. A transformational grammar framework for improvisation. In *First International Conference on New Music Concepts*, 2015.
- [322] Colin Raffel and Daniel P. W. Ellis. Optimizing dtw-based audio-to-midi alignment and matching. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 81–85, 2016.
- [323] Colin Raffel, Brian McFee, Eric J Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, Daniel PW Ellis, and C Colin Raffel. MIR-EVAL: A Transparent Implementation of Common MIR Metrics. In Wang et al. [399], pages 367–372.
- [324] Yves Raimond, Samer Abdallah, Mark Sandler, and Frederick Giasson. The Music Ontology. In Dixon et al. [118], pages 417–422.
- [325] Yves Raimond, Tristan Ferne, Michael Smethurst, and Gareth Adams. The BBC World Service Archive prototype. *Journal of Web Semantics*, 27–28:2–9, 2014.

Bibliography

- [326] Yves Raimond and Mark Sandler. Evaluation of the music ontology framework. In *The Semantic Web: Research and Applications: 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings 9*. Springer, 2012.
- [327] Dhanesh Ramachandram and Graham W. Taylor. Deep Multimodal Learning: A Survey on Recent Advances and Trends. *IEEE Signal Processing Magazine*, 34(6):96–108, 2017.
- [328] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.
- [329] Pedro Ramoneda Franco and Gilberto Bernardes de Almeida. Revisiting harmonic change detection. In *Audio Engineering Society Convention*, volume 149, oct 2020.
- [330] Preeti Rao, Hema A. Murthy, Ajay Srinivasamurthy, Rachel M. Bittner, Rafael Caro Repetto, Masataka Goto, Xavier Serra, and Marius Miron, editors. *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR 2022*, December 2022.
- [331] Christopher Raphael and Joshua Stoddard. Functional harmonic analysis using probabilistic models. *Computer Music Journal*, 28(3):45–52, 2004.
- [332] Sabbir M. Rashid, David De Roure, and Deborah L. McGuinness. A Music Theory Ontology. In *Proceedings of the 1st International Workshop on Semantic Applications for Audio and Music, SAAM ’18*, page 6–14, New York, NY, USA, 2018. Association for Computing Machinery.
- [333] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *EMNLP/IJCNLP*, pages 3980–3990. Association for Computational Linguistics, 2019.
- [334] Seungmin Rho, Seheon Song, Eenjun Hwang, and Minkoo Kim. COMUS: Ontological and Rule-Based Reasoning for Music Recommendation System. In *Advances in Knowledge Discovery and Data Mining*, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

-
- [335] Jenn Riley. Application of the Functional Requirements for Bibliographic Records (FRBR) to Music. In Bello et al. [20], pages 439–444.
- [336] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, and Douglas Eck. A hierarchical latent vector model for learning long-term structure in music. In *International conference on machine learning*, pages 4364–4373. PMLR, 2018.
- [337] Miguel Ángel Rodríguez-García, Luis Omar Colombo-Mendoza, Rafael Valencia-García, Antonio A Lopez-Lorca, and Ghassan Beydoun. Ontology-based music recommender system. In *Distributed Computing and Artificial Intelligence, 12th International Conference*, pages 39–46. Springer, 2015.
- [338] Francisco Jose Rodriguez-Serrano, Julio Jose Carabias-Orti, Pedro Vera-Candeas, and Damian Martinez-Munoz. Tempo driven audio-to-score alignment using spectral decomposition and online dynamic time warping. *ACM Trans. Intell. Syst. Technol.*, 8(2), oct 2016.
- [339] John Roeder. Pitch class, 2001.
- [340] Carles Roig, Lorenzo J Tardón, Ana M Barbancho, and Isabel Barbancho. Submission to MIREX: symbolic melodic similarity task, November 2013.
- [341] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.
- [342] J. Rothstein. *MIDI: A Comprehensive Introduction*. Computer music and digital audio series. A-R Editions, 1992.
- [343] Luke O. Rowe and George Tzanetakis. Curriculum learning for imbalanced classification in large vocabulary automatic chord recognition. In Lee et al. [239], pages 586–593.
- [344] Klaus-Jürgen Sachs and Carl Dahlhaus. Counterpoint, 06 2023.
- [345] Justin Salamon, Joan Serrà, and Emilia Gómez. Melody, bass line, and harmony representations for music version identification. In *Proceedings of the 21st International Conference on World Wide Web*, pages 887–894, 2012.

Bibliography

- [346] Amaia Salvador, Michal Drozdal, Xavier Giró-i Nieto, and Adriana Romero. Inverse cooking: Recipe generation from food images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10453–10462, 2019.
- [347] Emilio M. Sanfilippo and Richard Freedman. Ontology for Analytic Claims in Music. In Silvia Chiusano, Tania Cerquitelli, Robert Wrembel, Kjetil Nørnvåg, Barbara Catania, Genoveva Vargas-Solar, and Ester Zumpano, editors, *New Trends in Database and Information Systems*, pages 559–571, Cham, 2022. Springer International Publishing.
- [348] KC Santosh, Bart Lamiroy, and Laurent Wendling. DTW–Radon-based shape descriptor for pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 27(03):1350008, 2013.
- [349] Augusto Sarti, Fabio Antonacci, Mark Sandler, Paolo Bestagini, Simon Dixon, Beici Liang, Gaël Richard, and Johan Pauwels, editors. *Proceedings of the 24th International Society for Music Information Retrieval Conference, ISMIR 2023*, November 2023.
- [350] Edward W. Schneider. Course Modularization Applied: The Interface System and Its Implications For Sequence Control and Data Analysis. In *Association for the Development of Instructional Systems (ADIS), Chicago, Illinois, April 1972*, 1973.
- [351] A. Schoenberg, R.E. Carter, and W. Frisch. *Theory of Harmony*. University of California, 2010.
- [352] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics.
- [353] Abdul Shahid, Danny Diamond, and James McDermott. Patterns2KG: JAMS Pipeline for Modeling Music Patterns. In Antonis Bikakis, Roberta Ferrario, Stéphane Jean, Béatrice Markhoff, Alessandro Mosca, and Marianna Nicolosi Asmundo, editors, *Proceedings of the International Workshop on Semantic Web and Ontology Design for Cultural Heritage co-located with*

- the International Semantic Web Conference 2023 (ISWC 2023)*, Athens, Greece, November 7, 2023, volume 3540 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023.
- [354] Bidisha Sharma, Chitralekha Gupta, Haizhou Li, and Ye Wang. Automatic lyrics-to-audio alignment on polyphonic music using singing-adapted acoustic models. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 396–400, 2019.
- [355] Mohamadreza Sheikh Fathollahi and Farbod Razzazi. Music similarity measurement and recommendation system using convolutional neural networks. *International Journal of Multimedia Information Retrieval*, 10:43–53, 2021.
- [356] Saurabh Shrivastava. Bring rich knowledge of people, places, things and local businesses to your apps. Bing Blogs, July 2017. <https://blogs.bing.com/search-quality-insights/2017-07/bring-rich-knowledge-of-people-places-things-and-local-businesses-to-your-apps/>
- [357] Siddharth Sigtia, Nicolas Boulanger-Lewandowski, and Simon Dixon. Audio Chord Recognition with a Hybrid Recurrent Neural Network. In Müller and Wiering [286], pages 127–133.
- [358] Ian Simon, Dan Morris, and Sumit Basu. MySong: automatic accompaniment generation for vocal melodies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 725–734, 2008.
- [359] Federico Simonetta, Stavros Ntalampiras, and Federico Avanzini. Multimodal Music Information Processing and Retrieval: Survey and Future Challenges. In *2019 International Workshop on Multilayer Music Representation and Processing (MMRP)*, pages 10–18, 2019.
- [360] Federico Simonetta, Stavros Ntalampiras, and Federico Avanzini. Audio-to-score alignment using deep automatic music transcription. In *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6, 2021.
- [361] Elena Simperl and Markus Luczak-Rösch. Collaborative ontology engineering: a survey. *The Knowledge Engineering Review*, 29(1):101–131, 2014.

Bibliography

- [362] Jagendra Singh, Mohammad Sajid, Chandra Shekhar Yadav, Shashank Shekhar Singh, and Manthan Saini. A Novel Deep Neural-based Music Recommendation Method considering User and Song Data. In *2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1–7, 2022.
- [363] Amit Singhal. Introducing the Knowledge Graph: things, not strings. Google Blog, May 2012. <https://www.blog.google/products/search/introducing-knowledge-graph-things-not/>.
- [364] Camila Sitonio and Alberto Nucciarelli. *The impact of blockchain on the music industry*. Calgary: International Telecommunications Society (ITS), 2018.
- [365] William C. Sleeman, Rishabh Kapoor, and Preetam Ghosh. Multimodal Classification: Current Landscape, Taxonomy and Future Directions. *ACM Comput. Surv.*, 55(7), December 2022.
- [366] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. Embedding projector: Interactive visualization and interpretation of embeddings. *arXiv preprint arXiv:1611.05469*, 2016.
- [367] Temple F Smith, Michael S Waterman, et al. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197, 1981.
- [368] Daniel Stoller, Simon Durand, and Sebastian Ewert. End-to-end lyrics alignment for polyphonic music using an audio-to-character recognition model, 2019.
- [369] J.N. Straus. *Elements of Music*. Oxford University Press, 2021.
- [370] Bob Sturm, Maria Iglesias, Oded Ben-Tal, Marius Miron, and Emilia Gómez. Artificial intelligence and music: open questions of copyright law and engineering praxis. In *Arts*, volume 8, page 115. MDPI, 2019.
- [371] Bob Sturm, Joao Felipe Santos, and Iryna Korshunova. Folk music style modelling by recurrent neural networks with long short term memory units. In Müller and Wiering [286].

-
- [372] Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, and Mariano Fernández-López. The NeOn methodology for ontology engineering. In *Ontology engineering in a networked world*, pages 9–34. Springer, 2012.
- [373] Iris Sundin, Alexey Voronov, Haoping Xiao, Kostas Papadopoulos, Esben Jannik Bjerrum, Markus Heinonen, Atanas Patronov, Samuel Kaski, and Ola Engkvist. Human-in-the-loop assisted de novo molecular design. *Journal of Cheminformatics*, 14(1):1–16, 2022.
- [374] Christopher Sutton, Yves Raimond, and Matthias Mauch. The OMRAS2 Chord Ontology. <http://purl.org/ontology/chord/>, 2007.
- [375] Iman SH Suyoto and Alexandra L Uitdenbogerd. Simple orthogonal pitch with ioi symbolic music matching. In Downie and Veltkamp [121].
- [376] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society, 2016.
- [377] Kimmy Szeto. Ontology for Voice, Instruments, and Ensembles (OnVIE): Revisiting the Medium of Performance Concept for Enhanced Discoverability. *Code4Lib Journal*, 54, 2022.
- [378] Nazif Can Tamer, Yigitcan Özer, Meinard Müller, and Xavier Serra. High-resolution violin transcription using weak labels. In Sarti et al. [349].
- [379] David Temperley. Kostka-Payne corpus. <http://davidtemperley.com/kp-stats/>.
- [380] David Temperley. *The Cognition of Basic Musical Structures*. The MIT Press Series. MIT Press, 2001.
- [381] Nicholas Temperley. The Problem of Definitive Identification in the Indexing of Hymn Tunes. *Music Reference Services Quarterly*, 2(3-4):227–239, 1993.
- [382] Florian Thalmann, Alfonso Perez Carrillo, György Fazekas, Geraint A Wiggins, and Mark Sandler. The mobile audio ontology: Experiencing dynamic music objects on mobile devices. In *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, pages 47–54. IEEE, 2016.

Bibliography

- [383] The Real Book, 2004.
- [384] Matthias Thiemel. Dynamics, 2001.
- [385] Dan Tidhar, György Fazekas, Matthias Mauch, and Simon Dixon. TempEst: Harpsichord temperament estimation in a Semantic Web environment. *Journal of New Music Research*, 39(4):327–336, 2010.
- [386] Vibha Tiwari. Mfcc and its applications in speaker recognition. *International journal on emerging technologies*, 1(1):19–22, 2010.
- [387] Peter M Todd and Gregory M Werner. Frankensteinian methods for evolutionary music composition. *Musical networks: Parallel distributed perception and performance*, 3(4):7, 1999.
- [388] Luca Turchet, Francesco Antoniazzi, Fabio Viola, Fausto Giunchiglia, and György Fazekas. The internet of musical things ontology. *Journal of Web Semantics*, 60:100548, 2020.
- [389] Luca Turchet, Paolo Bouquet, Andrea Molinari, and György Fazekas. The Smart Musical Instruments Ontology. *Journal of Web Semantics*, 72:100687, 2022.
- [390] Luca Turchet, Johan Pauwels, Carlo Fischione, and György Fazekas. Cloud-smart musical instrument interactions: Querying a large music collection with a smart guitar. *ACM Transactions on Internet of Things*, 1(3):1–29, 2020.
- [391] Michael Uschold and Martin King. *Towards a methodology for building ontologies*. Citeseer, 1995.
- [392] Peter van Kranenburg, Berit Janssen, Anja Volk, et al. The Meertens tune collections: The annotated corpus (mtc-ann) versions 1.1 and 2.0. 1. *Meertens Online Reports*, 2016(1), 2016.
- [393] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.
- [394] Valerio Velardo, Mauro Vallati, and Steven Jan. Symbolic Melodic Similarity: State of the Art and Future Challenges. *Computer Music Journal*, 40(2):70–83, 06 2016.

-
- [395] Naresh N. Vempala and Frank A. Russo. An Empirically Derived Measure of Melodic Similarity. *Journal of New Music Research*, 44(4):391–404, 2015.
- [396] Hugues Vinet. The Representation Levels of Music Information. In Uffe Kock Wiil, editor, *Computer Music Modeling and Retrieval*, pages 193–209, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
- [397] Michail Vlachos, Dimitrios Gunopoulos, and George Kollios. Discovering similar multidimensional trajectories. In *Proceedings 18th International Conference on Data Engineering*, pages 673–684, USA, 2002. IEEE Computer Society.
- [398] Denny Vrandečić and Markus Krötzsch. Wikidata: A Free Collaborative Knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.
- [399] Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, editors. *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014*, October 2014.
- [400] Ziyu Wang, Ke Chen, Junyan Jiang, Yiyi Zhang, Maoran Xu, Shuqi Dai, Guxian Bin, and Gus Xia. POP909: A Pop-song Dataset for Music Arrangement Generation. In Cumming et al. [77].
- [401] David M Weigl, Tim Crawford, Aggelos Gkiokas, Werner Goebel, G Emilia, Nicol Guti, Cynthia CS Liem, and Patricia Santos. FAIR Interconnection and Enrichment of Public-Domain Music Resources on the Web. *Empirical Musicology Review*, 16(1):16–33, 2021.
- [402] Christof Weiß, Frank Zalkow, Vlori Arifi-Müller, Meinard Müller, Hendrik Vincent Koops, Anja Volk, and Harald G Grohgan. Schubert Winterreise dataset: A multimodal scenario for music analysis. *Journal on Computing and Cultural Heritage (JOCCH)*, 14(2):1–18, 2021.
- [403] Raymond P. Whorley and Darrell Conklin. Music generation from statistical models of harmony. *Journal of New Music Research*, 45(2):160–183, 2016.
- [404] Geraint Wiggins, Eduardo Miranda, Alan Smaill, and Mitch Harris. A framework for the evaluation of music representation systems. *Computer Music Journal*, 17(3):31–42, 1993.

Bibliography

- [405] Geraint Wiggins and Alan Smaill. Musical Knowledge: what can Artificial Intelligence bring to the musician? *Readings in Music and Artificial Intelligence*, 01 2000.
- [406] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016.
- [407] Peter Williams and David Ledbetter. Figured bass, 2001.
- [408] Thomas Wilmering, György Fazekas, and Mark B. Sandler. The Audio Effects Ontology. In de Souza Britto Jr. et al. [107], pages 215–220.
- [409] Anna Wolf and Daniel Müllensiefen. The perception of similarity in court cases of melodic plagiarism and a review of measures of melodic similarity. In J Wewers and U Seifert, editors, *Under Construction: Trans- and Interdisciplinary Routes in Music Research. Proceedings of SysMus11, Cologne 2011*, pages 215–222. Osnabrück Music, 2012.
- [410] Minz Won, Yun-Ning Hung, and Duc Le. A foundation model for music informatics, 2024.
- [411] Yiming Wu, Tristan Carsault, and Kazuyoshi Yoshii. Automatic chord estimation based on a frame-wise convolutional recurrent neural network with non-aligned annotations. In *27th European Signal Processing Conference, EUSIPCO 2019, A Coruña, Spain, September 2-6, 2019*, pages 1–5. IEEE, 2019.
- [412] Yiming Wu and Wei Li. Automatic Audio Chord Recognition With MIDI-Trained Deep Feature and BLSTM-CRF Sequence Decoding Model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(2):355–366, 2019.
- [413] Danai Xezonaki, Georgios Paraskevopoulos, Alexandros Potamianos, and Shrikanth Narayanan. Affective Conditioning on Hierarchical Attention Networks Applied to Depression Detection from Transcribed Clinical Interviews. In *INTERSPEECH*, pages 4556–4560. ISCA, 2020.

-
- [414] Qingyang Xi, Rachel M Bittner, Johan Pauwels, Xuzhou Ye, and Juan Pablo Bello. GuitarSet: A Dataset for Guitar Transcription. In Gómez et al. [158], pages 453–460.
- [415] Li-Chia Yang and Alexander Lerch. On the evaluation of generative models in music. *Neural Computing and Applications*, 32(9):4773–4784, 2020.
- [416] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 1317–1322. Ieee, 2016.
- [417] Yin-Cheng Yeh, Wen-Yi Hsiao, Satoru Fukayama, Tetsuro Kitahara, Benjamin Genschel, Hao-Min Liu, Hao-Wen Dong, Yian Chen, Terence Leong, and Yi-Hsuan Yang. Automatic melody harmonization with triad chords: A comparative study. *Journal of New Music Research*, 50(1):37–51, 2021.
- [418] Furkan Yesiler, Joan Serrà, and Emilia Gómez. Accurate and scalable version identification using musically-motivated embeddings. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 21–25. IEEE, 2020.
- [419] Zongyu Yin, Federico Reuben, Susan Stepney, and Tom Collins. Measuring When a Music Generation Algorithm Copies Too Much: The Originality Report, Cardinality Score, and Symbolic Fingerprinting by Geometric Hashing. *SN Computer Science*, 3(5), 2022.
- [420] Yuan Yuan, Yi-Ping Phoebe Chen, Shengyu Ni, Augix Guohua Xu, Lin Tang, Martin Vingron, Mehmet Somel, and Philipp Khaitovich. Development and application of a modified dynamic time warping algorithm (DTW-S) to analyses of primate brain expression time series. *BMC bioinformatics*, 12:1–13, 2011.
- [421] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor Fusion Network for Multimodal Sentiment Analysis. In *EMNLP*, pages 1103–1114. Association for Computational Linguistics, 2017.

Bibliography

- [422] Massimiliano Zanoni, Francesco Setragno, Augusto Sarti, et al. The violin ontology. In *Proc. of the 9th Conference on Interdisciplinary Musicology (CIM14)*. Citeseer, 2014.
- [423] Johannes Zeitler, Simon Deniffel, Michael Krause, and Meinard Müller. Stabilizing training with soft dynamic time warping: A case study for pitch class estimation with weakly aligned targets. In Sarti et al. [349], pages 433–439.
- [424] Yongwei Zhu, Mohan S Kankanhalli, and Sheng Gao. Music key detection for musical audio. In *11th International Multimedia Modelling Conference*, pages 30–37. IEEE, 2005.