ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

in cotutela con University of Luxembourg - Universitè du Luxembourg

DOTTORATO DI RICERCA IN

Law, Science and Technology

Ciclo 37

Settore Concorsuale: 01/B1 - INFORMATICA

Settore Scientifico Disciplinare: INF/01 - INFORMATICA

LEGIMATICS AND AI TOOLS FOR THE MONITORING OF EU LEGISLATION IN AGRIFOOD AND SDGS

Presentata da: ***Muhammad Asif***

Coordinatore Dottorato **Supervisore**

Monica Palmirani **Monica Palmirani**

Esame finale anno 2025

PhD-FSTM-2022-078
The Faculty of Science, Technology and Medicine

The Department of Legal Studies

DISSERTATION

Defence held in 2025 in Bologna

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG
EN INFORMATIQUE

AND

DOTTORE DI RICERCA
IN LAW, SCIENCE AND TECHNOLOGY

by

Muhammad Asif
Born on 10 October 1995 in Multan (Pakistan)

Legimatics and AI Tools for the Monitoring of EU Legislation in Agrifood and SDGs

Prof. Dr. Monica Palmirani,
dissertation supervisor
Professor, Alma Mater Studiorum – Università di Bologna

Prof. Dr. Leon Van Der Torre,
dissertation supervisor
Professor, Université du Luxembourg

Legimatics and AI Tools for the Monitoring of EU Legislation in Agrifood and SDGs

Muhammad Asif

2025

Dedication

To my beloved parents, what I am today is because of their efforts and care. Whose hands were always raised for my well-being.
Thank you, for standing by my side all these years.

Abstract

This work presents a comprehensive mechanism with algorithms for annotating norms, classifying EU legislation, and linking them to SDGs objectives. The dataset comprised of 15082 EU legislative documents in AKN file format that were adopted during the period of 1962-2021.

Complete work is divided into three tasks: (i) Detection and annotation of legal Definitions, (ii) Model design for classification of EU legislative documents to Goals and Targets of SDGs, and (iii) Linking EU legislative documents to Goals and Targets of SDGs.

The first task annotation of Delimiting Definitions is performed using Symbolic AI supported by LegalXML. For the purpose two independent Artificial Intelligence-based algorithms are designed for two different scenarios. These algorithms are implemented in Python using the ElementTree library and rule-based mining to annotate targeted text. The annotation is validated through indentation checks in the AKN format. The first algorithm annotates 899 documents, while the second algorithm annotates 1,272 documents. A total of 11,705 Definitions are successfully annotated in these documents.

For the second task, a new ML-based model is designed to link EU legislative to the Goals and Targets of SDGs. Based upon the literature review, two algorithms, SVM and KNN, were tried. SVM outperforms KNN with an accuracy of 53.34%, a weighted F-score of 70.04% and a macro F-score of 57.94% on the SDGs classification at the Goals level. At the Target level, SVM achieved 46.56% accuracy, 56.60% weighted F-score and 30.61% macro F-score.

In the third task, legislative text and annotated Delimiting Definitions are successfully linked with the Goals and Targets of SDGs using the model designed in the second task.

By integrating annotation, classification, and linking of EU legislation with SDGs, this research provides a robust mechanism for policymakers and researchers to monitor legislative alignment with SDGs objectives, enabling informed decision-making and effective policy formulation.

Keywords: Artificial Intelligence, AI & Law, SDGs, Machine Learning, NLP, Definition Annotation, SDGs Classification, SDGs Actions Linking

Acknowledgements

All the praises are for Almighty Allah, who is the most beneficent and merciful. First, I would like to say thanks to Almighty Allah, who created this universe and bestowed mankind and me with knowledge and wisdom. who invigorates me with the ability to accomplish this research work task and contribute a drop to the existing ocean of scientific knowledge.

I would not be able to find words to express my deepest gratitude and to acknowledge the efforts of my worthy supervisor` Professor Monica Palmirani (CIRSFID, University of Bologna, Italy) and Professor Leon Van Der Torre (University of Luxembourg, Luxembourg). They are just brilliant in their field of study, and they sketched this research and enabled me to achieve it practically. Without them, this research would have remained an unattainable aspiration for me, and their balance of care, encouragement and discipline motivates me to achieve my goals with precision and dedication.

I convey my special thanks to Dr Muhammad Aasim Qureshi for his moral support throughout my PhD journey and to my brother, Mr Muhammad Javed, and sisters, Mrs Sajida, Mrs Abida and Mrs Sidra Hashmat. Who always encouraged and helped me during this research work. They were always there to help whenever I needed any assistance and guidance. I want to say special thanks to my research colleagues and best friends Dr Davide Liga, Dr Réka Markovich, Dr Aasim Ali, Dr Muhammad Khurram Ehsan, Dr Michele Corazza, Dr Pere Pardo Ventura, Dr Christian Franck, Mr Asad Kamal, Mr Burhan ul Haq Zahir, Mr Affan Javed, Mr Naseer Ahmad, Mr Umar Shoukat, Mr Muhammad Bilal, Mr Aamir Ali, Mr Shahid Khan, Mr Muhammad Ahmad, Mr Bilal Shaid, Mr Aamir Hussain Qureshi, Mr Waqas Aslam and especially Mr Asghar Ali who were always there with me all the time to support me and to help me. Moreover, I cannot forget to say thanks to all my colleagues. I want to thank all my friends who prayed and encouraged my efforts.

Continuous moral support throughout my academic career enabled me to complete my research work efficiently and without worry. I love them all.

I want to express my heartfelt gratitude and remembrance of Mujahid Basheer (late). He was always overjoyed by my achievements and constantly kept me in his prayers. It deeply saddens me that he is not here to share this significant milestone with me, but his unwavering support and encouragement remain a source of inspiration. May ALLAH (S.W.T) grant him the highest rank in Jannah (AMEEN).

Finally, I would like to express my heartfelt gratitude for the financial support of my PhD, which has been generously funded by the PON grants of the Italian Government and the ERC HyperModeLex project.

Muhammad Asif

Table of Contents

List of Figures

List of Tables

List of ABBREVIATIONS

AI&LAW	Artificial Intelligence and Law
AI	Artificial Intelligence
NLP	Natural Language Processing
IR	Information Retrieval
ML	Machine Learning
DM	Data Mining
DL	Deep Learning
EU	European Union
UN	United Nations
FAO	Food and Agriculture Organization
AKN	Akoma Ntoso
SDGS	Sustainable Development Goals
UNO	United Nations Organization
IE	Information Extraction
DE	Definition Extraction
DA	Definition Annotation
DT	Decision Tree
SVM	Support Vector Machine
LT	Legal Texts
NB	Naïve Bayes
TF	Term Frequency
IDF	Inverse Document Frequency
URI	Uniform Resource Identifiers

ET	Element Tree
NBE	Knowledge-Based Extraction
WCL	World Class Lattices
LSTM	Long Short-Term Memory
POS	Part of Speech
ECHR	European Court of Human Rights
BOW	Bag of Words
PCA	Principal Component Analysis
OCR	Optical Character Recognition
CFR	Creditor Reporting System
CSV	Comma-Separated Values
Acc	Accuracy
W.P	Weighted Precision
W.R	Weighted Recall
W.F	Weighted F-score
M.P	Macro Precision
M.R	Macro Recall
M.F	Macro F-score
OASIS	Organization for the Advancement of Structured Information Standards

The technical terms such as "*Definition* and *Delimiting Definition*" are italicised to enhance the reader's understanding.

Chapter 1

1 Introduction

The term 'Sustainable Development' was first sanctioned in 1987 by the Brundtland World Commission [1] also known as the World Commission on the Environment and Development [2]. It combines the concepts of 'sustainability' and 'development' to emphasize the need for long-term global well-being and growth for future generations [3]. In simplest terms, economic growth and development should not come at the cost of wounding the current environmental situation, where the resources are oppressed and exploited beyond the capacity for renewal [4]. Over the years, these indicators of sustainability and development have advanced, leading to the creation of an extended action plan aimed at achieving sustainable development on a global scale [5]. This plan, focused on ensuring the planet's long-term sustainability, was adopted by the United Nations [6].

The United Nations (UN) member states have taken significant steps towards global sustainability [2], leading to the introduction of the Sustainable Development Goals (SDGs) on September 25, 2015 [7]. These Goals are part of the 2030 Agenda for Sustainable Development, aimed at ensuring peace and prosperity for the planet [8]. A total of 193 member countries agreed on a set of 17 Goals of SDGs [9], with 169 associated Targets [10], designed to create a prosperous future for current and future generations. These Goals are universally applicable to all nations, regardless of geographic location or Gross Domestic Product (GDP) [11][12].

This ambitious agenda for global sustainability serves as the world's roadmap for addressing a wide range of complex challenges [13]. This includes poverty, hunger, health & well-being, and the pursuit of peace

& justice for all [14]. It also acknowledges critical issues such as combating climate change, providing quality education, protecting biodiversity, promoting economic growth, reducing social inequalities, building sustainable cities & communities, and ensuring responsible production & consumption of goods [15] [16].

One of the key agendas of the Sustainable Development Goals (SDGs) is to achieve 'No Poverty' [17], 'Zero Hunger' [18], and 'Good Health & Well-Being' [19]. The Food and Agriculture Organization (FAO) of the UN has passed resolutions to protect food security and enhance the agricultural sector to address hunger and poverty, thereby contributing to humanity's overall well-being [20]. The European Union (EU) is also collaborating with the FAO within the framework of the SDGs, enacting legislation and resolutions [21] to support these Goals [22].

To quantify the relationship between EU legislation and the SDGs, a Machine Learning-based algorithm is employed to design a ML-model. This model computes and aligns the representation of each of the SDGs goals and targets (i.e. 17 Goals and 169 Targets) with the dataset of EU-legislative files [23][24]. This ML model not only classifies text data but also offers several advantages, including the ability to identify trends and patterns within legislative documents [25].

Further, this research provides a comprehensive framework that enhances the monitoring and evaluation of legislative alignment with global sustainability objectives. This integration facilitates a structured approach for policymakers, regulatory bodies, and researchers to assess whether current and proposed legislation effectively supports sustainable development initiatives.

1.1 Scope and Targets

This study focuses on Legimatics and AI tools for monitoring EU legislation in agrifood and SDGs. The study integrates symbolic-AI, NLP, and ML to automate legal norms' extraction, annotation, and classification within EU legislative documents.

1.1.1 Scope

The primary scope of this thesis is to evaluate the efficiency and effectiveness of European Union legislation in advancing and achieving the targets of the Sustainable Development Goals. To achieve this, the study focuses on Directives and Regulations from the European legislation, which are annotated to extract definitions and identify potential inconsistencies. A classification model is developed to link these legislative documents and their annotated definitions to the relevant Goals and Targets of the SDGs. Other types of EU legal instruments—such as Decisions, Recommendations, Declarations, and Resolu-tions—are not included in this analysis. In the dataset, Regulations are labelled with the prefix 'R' and Directives with 'L'.

1.1.2 Targets

The target of this study are multiple as expressed below

T1-Auto-detection, annotation, and extraction of legal norms

T2-Algorithm Design for the classification of legislative documents

T3-Linking of EU legislation with SDGs

Details can be seen in the following sub-sections.

1.1.2.1 T1-Auto-Detection, Annotation, and Extraction of Legal Norms

The target T1 is to detect the Delimiting type of Definitions from the European Union legislation using the symbolic AI technique—Rule-Based Mining. After the successful detection of all the definitions present in the EU legislation. All detected definitions are annotated with metadata that includes definition heading and definition body, i.e., the def tag is added with the 'Definiendum' that is the heading of the definition and the defBody tag is added with the body of the definition. A unique identification number (eId) is added within the def tag and defBody tag, i.e., " < def eId = "def_1" > applicant authority </def > " and < defBody eId = "defBody_1" >

means a central liaison office; </defBody > with each definition by leveraging LegalXML and symbolic AI methodologies. T1 aims to identify and structure legal norms, ensuring that legislative documents are systematically processed for regulatory insights.

1.1.2.2 T2-Algorithm Design for the Classification of Legislative Documents

T2 is based on developing an ML-based classification model to classify the EU legislative documents into the Goals and Targets of the SDGs. For the Purpose, Support Vector Machine and K-Nearest Neighbor are applied to develop a classification model based in their better performance. That model will be saved and used to classify the EU legislative documents with the SDGs.

1.1.2.3 T3-Linking of EU legislation with SDGs

The third target is to identify the actions related to the SDGs in EU legislation and to link with them. For the purpose, ML-based SVM-model that is used in T2 for the classification of SDGs will be employed to link legislative documents and extracted definitions according to their relevance to the SDG objectives for facilitating a structured understanding of how EU legislation aligns with global SDGs: at the Goals and Targets Level. For example: SDG 16 (Peace, Justice, and Strong Institutions) is linked to 610 legislative files, whereas 2,910 definitions are linked to SDG 16 (Peace, Justice, and Strong Institutions), making it the most frequently addressed goal in legislative definitions.

Ultimately, this work aims to contribute to the field of legal informatics by proposing a scalable and efficient AI framework for analyzing, categorizing, and monitoring legislative developments, thereby supporting policymakers, legal practitioners, and researchers in navigating complex regulatory landscapes.

1.2 Motivation

The European Union and the Food and Agriculture Organization collaborate to preserve food to address global hunger issues under the UN's SDGs agenda. As a key global actor, the European Union has enacted numerous legislative measures to support the SDGs, particularly in agriculture, food safety, and resource distribution. These measures aim to enhance agricultural production and improve handling while balancing security, quality, quantity, and the equitable distribution of food resources in relation to a sustainable market. These policies (i.e., Norms in EU legislation) show some inconsistencies; for instance, one of the 17 SDGs is addressed in three out of ten legislative documents. In contrast, another SDG is discussed in only one out of twenty legislative documents. To overcome these inconsistencies and to in-line the policies to monitor their efficiency compared with the SDGs [26] there is a need for a systematic and scalable mechanism to evaluate whether and how EU legal instruments align with the SDGs at the level of specific Goals and Targets.

This gap in monitoring and alignment has serious implications. Without clear insights into how EU legislation supports specific SDG Goals and Targets, policy actions may be fragmented, inconsistent, or even counterproductive. For instance, if one SDG is well-integrated across several legal texts while another is barely addressed, this imbalance can lead to misallocation of resources, ineffective interventions, and missed opportunities for achieving holistic sustainability. Furthermore, the lack of transparency and traceability in how legal definitions and actions correspond to SDG objectives limits the ability of lawmakers, researchers, and citizens to evaluate progress and propose meaningful improvements.

This work is therefore motivated by an urgent need to bridge this policy-intelligence gap. By developing AI-based systems to extract legal definitions, classify legislation in relation to the SDGs, and link legislative actions to SDG targets, this research introduces mechanism that

makes EU legislation more interpretable, analyzable, and accountable in the context of global sustainability efforts.

The development of this research is critical to prevent continued gaps in assessing how legislative actions align with established sustainability objectives. Omitting this, the EU and other policymaking bodies will continue to operate without a reliable mechanism to assess legislative alignment with sustainability goals. This could result in continued inefficiencies, poor accountability, delayed SDG implementation, and ultimately, failure to meet the 2030 Agenda. The consequences would not only be bureaucratic but also social and environmental, impacting food security, equitable development, and the credibility of international sustainability commitments.

In this context, the proposed research is not only innovative but also necessary to achieve the agenda of the United Nations of the Sustainable Development Goals by 2030. It empowers legal and policy communities with intelligent tools to align legal norms with global priorities, enabling better governance and more effective responses to the complex challenges of our time.

1.3 Objective

The primary aim of this thesis is to detect, annotate and extract the norms from EU legislation and to establish a methodology to design a decision-making support system that classifies the detected norms into SDGs. The objectives of this study are as follows:

- **Obj-1:** To detect and annotate the norms (definitions) in the EU legislation
- **Obj-2:** To establish a model to classify the legislative documents into the SDGs at the Goals and Targets Levels
- **Obj-3:** To quantify the actions found in EU Legislation with SDGs (Goals and Targets)

1.4 Problem Statement and Research Questions

Hunger is the world's most considerable solvable problem. The United Nations Organization is working keenly on this matter. The Food and Agriculture Organization (FAO) is working under the UN to overcome hunger issues. FAO passed resolutions to protect the food and enhance the agriculture sector. The European Union is also working in collaboration with FAO under the umbrella of SDGs. Legislation on Agrifood can handle the important challenges to overcome hunger issues, i.e., handling and balancing quality, security, and equal distribution of food resources in contrast with the market's sustainability. The EU adopts legislation to support the SDGs to address hunger-related challenges and overcome hunger issues. There were some inconsistencies found in norms, so there is a need to help agrifood operators in legal compliance and the detection of inconsistent norms and rules to monitor their efficiency with the SDGs. Moreover, this thesis aims to monitor their efficacy in facilitating decision-making & policymaking and identify the actions linked to the SDGs.

The problem statement of this work is as follows: it is to detect, annotate and extract the norms in the EU legislation and to classify the EU legislative documents into the SDGs at the Goals and Targets level. Finally, quantification of actions found in the EU legislation in the SDGs.

The research questions that will be addressed in this thesis are as follows:

1. **RQ1:** How can norms be detected, annotated, and extracted from EU legislative text files?
2. **RQ2:** How can a classification model be developed to classify EU legislative text into the Sustainable Development Goals?

3. **RQ3:** How can the efficacy of rules and norms be monitored to facilitate decision-making and policy formulation while identifying actions linked to the SDGs?

1.5 Key Contributions

The Key Contributions of this study are:

i. The automatic detection of Norms using an Artificial Intelligence-based algorithm.
ii. The auto-annotation of Delimiting Definitions in the EU legislative AKN files.
iii. Develop a Machine Learning-based model to classify the Sustainable Development Goals.
iv. Classify the 17 Goals and 169 Targets of the SDGs using an ML-based model.
v. Linking the actions taken by the European Union in EU legislation at the Goals and Targets Level of the Sustainable Development Goals.
vi. Linking the extracted Delimiting Definition with the Sustainable Development Goals at the Goals and Targets Level to inline the policies with the SDGs.

This study helps legislators to identify and address inconsistencies in normative references across different SDGs. For instance, while some Goals are referenced with high frequency, others receive minimal attention. This imbalance underscores the need for a more equitable focus on all SDGs. For example, SDG 8 (Decent Work and Economic Growth) is linked to 1278 definitions in the legislative files, whereas SDG 6 (Clean Water and Sanitation) appears only in 11 definitions. This discrepancy highlights the necessity for policymakers to strengthen legislative efforts related to SDG 6.

Similarly, when analyzing the linking of Delimiting Definitions within EU legislation, this study revealed inconsistencies in how norms align with SDGs. For example, 2,910 definitions are linked to SDG 16 (Peace, Justice, and Strong Institutions), making it the most frequently addressed

goal in legislative definitions. In contrast, SDGs 5 (Gender Equality), 6 (Clean Water and Sanitation), and 15 (Life on Land) are associated with only 12, 10, and 11 definitions, respectively. These findings emphasize the need to enhance legislative attention toward these underrepresented goals, ensuring a more balanced approach to sustainable development policymaking. Figure 1 presents a conceptual mapping representation of the thesis objectives, their corresponding research questions (RQ), and the outcomes associated with each objective and its respective research question.

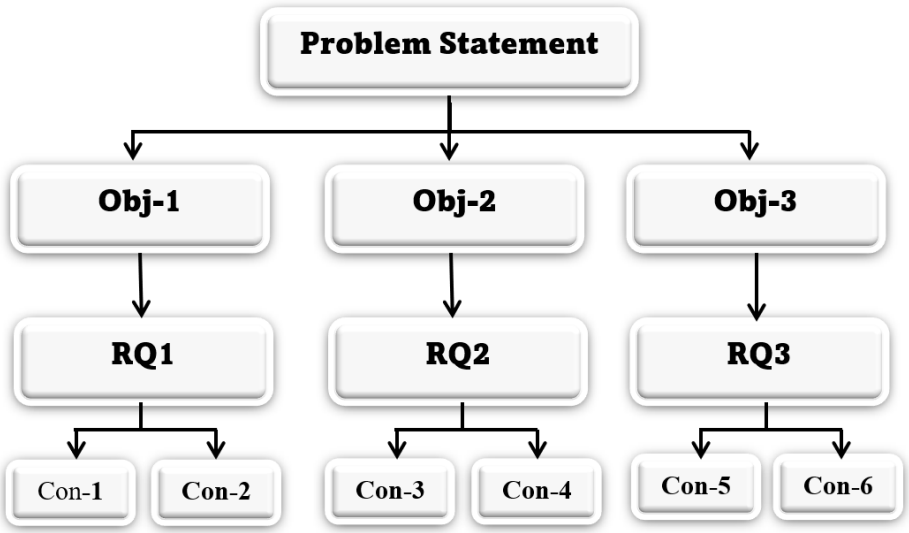

Figure 1: Conceptual Mapping of Thesis Objectives, Research Questions, and Outcomes

The rest of the thesis is organized into six Chapters. The second chapter presents the Legislation Modelling: Eur-Lex and Sustainable Development Goals, and the third chapter is about the Normative Definition Anatomy, in which the detection, annotation and extraction of the legislative definitions are presented in detail. Chapter four is about the classification of Sustainable Development Goals in the classification EU Legislation and Legislative Definitions into the Goals and Targets of the SDGs are presented and Chapter five is about Linking the Actions of EU

Legislation with SDGs. Chapter 6 provides a detailed analysis of the process of linking definitions with the Sustainable Development Goals (SDGs), and finally, the thesis is concluded by outlining potential directions for future research and development.

Chapter 2
Legislation Modelling: Eur-Lex and Sustainable Development Goals

2 Introduction

The European Union adopts legislation through various legislative procedures. After the legislative body's final decision and once it is signed by the President and General Secretaries of the European Parliament and the European Council, the legislation is published in the Official Journal [27]. The publication of EU legislation in the Official Journal serves as a foundation for its accessibility and organization, further facilitated by platforms like EUR-Lex and detailed naming conventions for legislative documents [28].

2.1 Eur-Lex

EUR-Lex is the official online gateway of the European Union Law. It provides official and most comprehensive access to EU legal documents [29]. EUR-Lex served as a vital repository for EU Legislation, case law, and other public documents. You can find legislation, case law, and various public documents on EUR-Lex. These documents are freely available and accessible in all 24 official EU languages. These official documents on EUR-Lex are updated on a daily basis [30].

2.2 ELI

ELI stands for ‘European Legislation Identifier’ [31]. It makes legislation available online in a standardized format officially published across Europe. Using ELI, legislative documents can easily be accessed, reused, and exchanged across borders [32]. ELIs use HTTP URIs (Uniform Resource Identifiers), which are readable for both humans and computers [33].

Examples

The structure of the European Legislation Identifier is shown below:

http://data.europa.eu/eli/{typeOfDocument}/{yearOfAdoption}/{numberOfDocument}/oj

2.3 CELEX

CELEX is a documentation system for European documents produced by the EUR-OP (Office of the Official Publications of the European Communities). The system assigns a unique identifier or reference number to EU documents, independent of the document's language. Most of the documents are assigned a CELEX number [34].

An EU document is assigned a CELEX number consisting of different parts that vary slightly depending on the document type. In CELEX, the most common cases used for identification are [35] [36]:

i. Sector
ii. Year
iii. Document type
iv. Document number

2.3.1 Sector

Documents in EUR-Lex fall into one of 12 sectors.

0 Consolidated texts
1 Treaties
2 International agreements
3 Legal acts
4 Complementary legislations
5 Preparatory documents
6 EU case-law
7 National transposition
8 References to national case law concerning EU law
9 Parliamentary questions
E EFTA documents
C Other documents published in the Official Journal C series

2.3.2 Year

The year in the CELEX usually describes the year in which the document was published/adopted.

2.3.3 Document Type

In CELEX, the added descriptor usually has 1 or 2 letters describing the document type. In this study, the legal acts of the European Union are used for the detection and annotation of *Delimiting Definition*s. Legal acts usually use these three descriptors[36]:

i. D: Decisions
ii. R: Regulations
iii. L: Directives

Directives (L) and Regulations (R) are incorporated in this study. So, in our dataset, only two letters, L and R, will be found with file names in the dataset.

2.3.4 Document Number

The CELEX system uses the last four digits that appear at the ends of the file name to represent the document number.

The following are the examples [37]: one for the Directive and one for the Regulations are given below.

32010L0024: consists of 4 parts:

- 3 is for the sector (for legal acts)
- 2010 is the year of the document publishing
- L stands for Directive and shows the type of document
- The last four digits, i.e., 0024, are the number of the document

32010R1015: consists of 4 parts:

- 3 is for the sector (for legal acts)
- 2010 is the year of the document publishing
- R stands for Regulation and shows the type of document
- The last four digits, i.e., 1015, are the number of the document

2.4 Formex

Formex stands for ‘Formalized Exchange of Electronic Publications’. These legislative documents are based on international XML standards on the recommendation of W3C on 10 February 1998 [38]. It describes the legislation format for the data exchange between the contractors and the publication office [39]. It supports the documentation in English only [40].

2.5 Akoma Ntoso

Akoma Ntoso, also known as AKN, means "linked hearts" [31]. The Akan people of West Africa commonly use this term to signify agreement and understanding. AKN is a knowledge-oriented management system for normative documents that uses XML markup standards. It establishes common standards that enable parliaments to exchange information and provide open access to parliamentary documentation [41]. Using an XML schema, it provides a structured representation of legal and legislative documents.

AKN was first developed in 2004 by the UNDESA (United Nations Department of Economic and Social Affairs) to develop Africa i-Parliament Action Plan [42]. The aim was to digitalize legislation and public access towards the legislative text of the African Parliament. From 2011, the AKN gained broader international recognition beyond Africa, particularly across America and Europe, where governments and legal informatics communities saw its potential in digitizing legal workflows [43]. AKN is a global legal XML standard approved by the OASIS (Organization for the Advancement of Structured Information Standards) Legal Document ML (LegaldocML) Technical Committee body in 2013. In 2018, OASIS made AKN an official open standard for legal document structuring [44].

The European Parliament uses AKN4EU to represent legislative documents. AKN4EU version 3.0 was adopted on 17 April 2020 to represent

EU legislative documents. AKN4EU is a Common Structured Format for EU Legislative Documents. It is a technical markup standard developed by EU institutions based on Akoma Ntoso (AKN). AKN4EU is used to represent legislative, judicial, and executive documents in a structured manner [45] [46].

Nowadays, AKN is used in more advanced ways by using AI-driven legal research by combining symbolic AI and Natural Language Processing. It is used in automated ways to extract norms from the legislative text, for legal text classification, summarization of legal provisions, and in machine-assisted legal support. Projects like EU's AI4Law and Legimatics leverage Akoma Ntoso to enhance machine-readable legal ontologies.

2.6 Sustainable Development Goals

In 2015, the United Nations introduced the SDGs as a global call to action to end poverty, protect the planet, and ensure prosperity for all by 2030. The SDGs are a set of 17 goals adopted by the United Nations member states to promote global sustainability. These ambitious goals are interconnected to address a wide range of social, economic, and environmental challenges, such as ensuring quality education, promoting good health and well-being, achieving gender equality, and eliminating hunger. The SDGs provide a comprehensive plan for creating a sustainable and equitable world. Figure 2 illustrates the representation of the 17 Goals of the SDGs.

These 17 goals encourage collaboration across borders for a brighter future and a sustainable world. Each SDGs includes several targets designed to achieve the sustainability agenda. A total of 169 targets were introduced, accompanied by 241 indicators. This ambitious agenda for global sustainability serves as the world's roadmap to cohesively address complex challenges, including combating climate change, promoting economic growth, building sustainable cities and communities, and ensuring the responsible production and consumption of goods.

Figure 2: The 17 Goals of SDGs of the UN

Chapter 3
Normative Definition Anatomy

3 Introduction

In recent years, data volumes have been growing at an exponential rate. Traditional tools and techniques are no longer sufficient to process, refine, and extract essential & meaningful information from this vast data landscape. The concept of big data has emerged as a solution to manage and analyze such large datasets [47]. This data can be either structured, such as credit card numbers, dates, geolocations, addresses, and stock information or unstructured, including spreadsheets, emails, text files, surveillance footage, images, videos, survey reports, and machine-generated formats. Text data, a prime example of unstructured data, is typically analyzed to identify patterns that can derive informed actions [48].

The vast amount of unstructured text data makes it practically impossible to extract structured content and patterns manually. Numerous efforts have focused on automating this extraction process [49][50][51]. Significant progress has been made in the field of Text Mining [52][53][54] and Computational Linguistics [55] [56] [57]. In text mining, Information Retrieval (IR) techniques are applied to extract and preprocess information [58]. Information Extraction (IE) is one of the most effective techniques to extract meaningful information from text, though it is considered one of the challenging tasks in the domain of Natural Language Processing (NLP) [59].

IE is a way to extract structured and meaningful information from semi-structured or unstructured text data. Relationship extraction is one of the techniques of IE in NLP that deals with identifying and extracting relationships between the entities found in the text. Co-references-based extraction is a way to identify the different expressions that refer to the same entity in the text. In IE, some techniques address domain-specific problems to manage their unique characteristics effectively. One such domain-specific technique is Definition Extraction (DE), which deals with the extraction and identification of terms' definitions or phrases from the text [60][61]. Event extraction deals with extracting and

identifying actions or events in the text. The identification of attributes and characteristics falls under the umbrella of attribute extraction. Key phrase extraction deals with extracting and identifying meaningful and short phrases that represent the core themes and ideas in the text.

DE is a technique for identifying and extracting Definitions, terms, or phrases from the text. Different approaches can be used for the extraction of *Definition*s, such as rule-based mining, graph-based mining, pattern-matching, and hybrid approaches. The rule-based approach relies on pre-defined linguistic rules and syntactic structures to identify the specified text. Figure 3 shows the hierarchical representation of information extraction techniques.

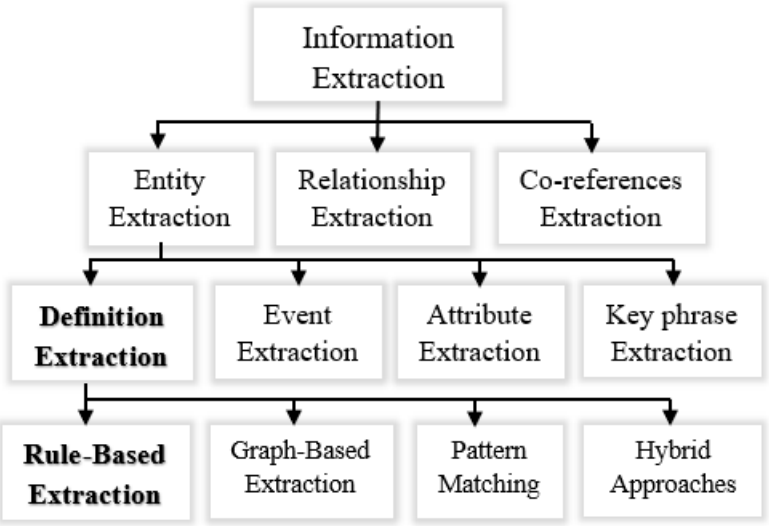

Figure 3: Hierarchal Representation of Information Extraction Techniques

The following subsection thoroughly discusses the target of this research, i.e., the definition, the definition in legislation, different parts of the definition, different types of definitions in legislation, and related topics.

3.1 Definition

According to Aristotle, a *Definition* is “a phrase signifying a thing's essence” [62]. With the advancement of modern ontological approaches,

a *Definition* can be defined as “the degree of distinctness in the outline of an object” [63]. The *Definition* is conceived as commitments which need supporting arguments [64]. *Definition*s are a powerful and substantial constituent in Legislation [41]. If it examines how legal arguments are made, it may see that *Definition*s are contested, upheld, refuted, and successfully backed by various arguments [65]. In the legislative text, the term's meaning is controlled by the *Definition*s [66].

3.2 Definition Extraction

Definition Extraction is the process of identifying and extracting Definitions or phrases from the text [67]. Various automated and semi-automated techniques and algorithms are available for extracting Definitions, significantly reducing the effort and time required compared to manual extraction [68][69]. In recent years, researchers have made considerable progress in DE with varied levels of success [70][71][72][73]. DE has proven valuable across multiple domains, including customer support automation, automatic summarization, ontology engineering, knowledge graphs, e-learning, automated documented summarization, legal analysis, and lexicon or glossary construction. Recently, DE has emerged as a key topic in the legal field, particularly in detecting and extracting Definitions from legal texts [74][75]. However, most DE work, to date, has relied on statistical methods, with relatively few studies focusing specifically on legal aspects [73][74]. This highlights a need for further research on extracting Definitions tailored to legal contexts.

3.3 Definition in Legislation

The *Definition* has crucial importance in legislation [76]. It constitutes the fundamental premise of arguments from classification. *Definition*s have different subject matters in the legislative text. *Definition*s in the legislative text are used for stipulations and to explain what a phrase means in the text [77]. The content of the legislative text is never

reiterated. The *Definition* is only necessary if the term's meaning is not understandable or if it has a technical or scientific implication. In this regard, the *Definition* will help to understand the meaning of the targeted term [78]. It should extend the meaning of the term. *Definition*s are not needed to define simple or less technical terms. *Definition*s are used in legislation only if they can improve the comprehensibility of a legislative text. In the legislative text, it can only be used for the following reasons [79][80].

- To keep the text short and easy to read and understand
- To avoid any ambiguity regarding the meaning of the word
- To explain the meaning of a new or unusual word
- To make the text shorter by avoiding repetition
- To define new legal concepts (constitutive role of the legal *Definition*s)
- To make the *Definition* linkable in a unique way, especially in the case-laws

In Legislation, the *Definition*s should also be used to avoid uncertainty:

- Usually, a term has no precise or consistent meaning
- A term has different meanings, and it might be difficult to determine which one is indicated
- The term's ordinary meanings cover many things, but only specific ones are indicated

Often in the legal domain, there are *Definition*s conditioned by jurisdiction (e.g., non-applicable to Malta) by a temporal parameter (e.g., applicable only during the COVID), by a specific geographic situation (e.g., not applicable in the airport). The legal *Definition*s also define the equivalence between different terms emerging over time due to the evolution of society (e.g., mother and father are now equivalent to parent). The legal *Definition*s also define a relationship of belonging and identity when a term is used instead of a whole class (e.g., bike count-as vehicle).

3.3.1 Parts of Definition

A *Definition* is composed of the following four components: [81][82] [83].

- Definiendum: subject of the *Definition*, also called term
- Definitor: a verb phrase which is used to introduce the *Definition*, i.e. “means” or “is”
- Definiens: phrase which is used to define something in the *Definition*
- Condition: condition connected with situations, temporal parameters, geographic areas, etc.

For example, let’s have the following short text [84]:

> "ship" means any seagoing vessel, whether publicly or privately owned, which is ordinarily engaged in commercial maritime operations. Fishing vessels are not included in this definition.

In the text given above, *Definition* of “ship” is explained. The “ship” is Definiendum which is the subject of the *Definition*, ‘means’, verb phrase, is Definator, ‘any seagoing vessel, whether publicly or privately owned, which is ordinarily engaged in commercial maritime operations’ is a Definiens which is used to define “ship” and ‘Fishing vessels are not included in this *Definition*,’ is Condition, the supplementary phrase.

There are different types of *Definition*s to define technical terms in the legislation. The section below discusses the types of *Definition*s used in legislation. The example [84] Figure 4 indicates the different parts of the *Definition*s (Definiendum, Definitor, Definiens, and rest/conditions).

> "ship" means any seagoing vessel, whether publicly or privately owned, which is ordinarily engaged in commercial maritime operations. Fishing vessels are not included in this definition,

Figure 4: Parts of Definition

3.3.2 Types of Definitions in Legislation

In legislation, diverse definitions are used to define ambiguous terms. These types of *Definition*s are categorized into the following:

1. Delimiting Definition
2. Extending Definition
3. Narrowing Definition
4. Count-as Definition
5. Mixed Definition

Details can be seen in the below sub-section.

3.3.2.1 Delimiting Definitions

Delimiting Definition is introduced when there is a need to determine the boundaries and limits of the defining term in the legislation. These types of information usually start with the word ‘means’ [85].

For example, let’s have the following short text [86]:

> "energy-related product" means any good having an impact on energy consumption during use, which is placed on the market and/or put into service in the Union, including parts intended to be incorporated into energy-related products covered by this Directive which are placed on the market and/or put into service as individual parts for end-users and of which the environmental performance can be assessed independently;

In the text given above, the definition of “energy-related product” is explained in which the “energy-related product” is Definiendum which is the subject of the *Definition*, ‘means’, verb phrase, is Definator, ‘any good having an impact on energy consumption during use, which is placed on the market and/or put into service in the Union, including parts intended to be incorporated into energy-related products covered by this

Directive which are placed on the market and/or put into service as individual parts for end-users and of which the environmental performance can be assessed independently' is a Definiens which is used to define "energy-related product".

Different types of definitions are used to define technical terms in legislation. The section below discusses the types of definitions used in legislation.

3.3.2.2 Extending Definitions

Extended Definition is usually used to extend the meaning of the term that does not have. It can also be said that it is used to explain the original term used in Legislation. These types of *Definition*s usually start with the word 'includes' [87].

For example, let's have the following short text [37] [88]:

> "name of the device" includes any user-provided data or information that would help to differentiate the device from others of a similar kind of device.

In the text given above, the 'name of the device' is going to be distinguished. The name of the specific device can be distinguished by including any user-provided data or information that would help to differentiate the device from others of a similar kind of device.

3.3.2.3 Narrowing Definitions

*Narrowing Definition*s narrow or restrict the term's ordinary meaning by setting limits. In *Narrowing Definition*, the term is restricted within the boundaries. It usually starts with the word 'means'. The second part that starts with 'or' does not the part of the *Narrowing Definition* [89].

For example, let's have the following short text [90]:

> "dealer" means a retailer or other person who sells hires, offers for hire-purchase, or displays products to end-users.

"retailer" does not include someone who sells or displays products to end-users.

In the text given above, the term is restricted within the boundaries. The ‘dealer’ is limited in the boundary who sells, hires, offers for hire-purchase, or displays products to end-users. Whereas the ‘retailer’ is narrows as someone who sells or displays products to end-users.

3.3.2.4 Count-As Definitions

*Count-As Definition*s are often used in legislation to ensure that rules can apply to non-traditional or evolving situations. This type of definition is introduced in legislation to specify actions, items, or conditions to be treated as part of a specific category, even if they don't traditionally fit within that category. Essentially, they help adapt legal terms to new or changing contexts. In this case, the legal definition defines a relationship of belonging to a general class [91].

For example, let’s have the following short text [92]:

> A pollutant is defined as any substance or energy introduced into the environment that causes harm. This includes chemical substances, heat, light, noise, and biological materials.

In the text above, the definition of pollutant is explained, with the pollutant as the heading and the rest of the definitions as the body. *Count-as Definition*s are particularly common in electronic and intellectual property law.

3.3.2.5 Mixed Definitions

A type of *Definition* is necessary to set limits and clarify elements excluded or included from the term. This type of *Definition* usually starts with the word ‘means’ and has the word includes [93].

For example, let’s have the following short text :

> "Heavy motor vehicle licence" means an HMV class licence, which permits the holder to drive all types of heavy vehicles i.e., buses, trucks and other commercial vehicles including light commercial vehicles.

In the given above text, the HMV licence holder can drive light and heavy commercial vehicles. In the legislation, a *Definition* can be drafted besides using means and includes words. In Legislation, a *Definition* can be referred to as an act. It can also be categorized into sections, subsections, subsections, chapters, sub-chapters, headings, subheadings and sub-subheadings.

This study targeted the extraction and annotation of *Delimiting Definition*s from the EU legal texts. Definitions can be conceived as commitments that need to be supported by arguments when questioned. Evaluation of *Definition*s involves examining the entire justification offered for them. The legal *Definition* can be examined from three perspectives:

1. By their pragmatic function
2. By their Propositional structure
3. By their Argumentative Role

3.4 Types of Definitions in Law

In Legislation, *Definition*s are not used for simple propositions. They can define or describe a specific term. In law, *Definition*s can be categorized into three types. The rest of the details are discussed in the next sections.

3.4.1 Pragmatic Function

Definition in law may be used for a variety of pragmatic purposes. It can be used in law to define a new meaning of a term, or it can also be used to describe the term. At the pragmatic level, the *Definition*s can be divided into two classes: (i) The *Statutory Definition*s and (ii) The *Descriptive Definition*s. In law, Definitions are used for two basic purposes:

one is for assigning a specific meaning to any term, and the second is for explaining the meaning of an ambiguous and unclear term.

3.4.1.1 Statutory Definitions

*Statutory Definition*s are used to relate the definitional discourse with their propositional content. It simply discourses the particular definition of the word. The *Statutory Definition* helps both parties and commits them to the definition of any specific term. It commits the legislator and the people subject to the law to a particular term or the definition of the particular term. The *Statutory Definition* is used to clarify "what the word means" in a descriptive manner. It describes the actual meaning of the word used in the legislation or defines the meaning of the verb which carries the actual meaning. *Statutory Definition*s are used to avoid conflicts due to word ambiguity.

3.4.1.2 Descriptive Definitions

Descriptive Definitions describe the actual meaning of the word used in the legislation. *Descriptive Definition*s are usually used to define technical terms. These definitions are used to clarify and solve conflicts of opinion.

3.4.2 Propositional Structure

*Propositional Definition*s are made up of different propositions and connections between them. It is used for the declaration of terms that are either true or false, not in both states. Propositional law is always defined with the help of declarative sentences. In Law, a term can be defined differently according to the legislator's purpose. At the propositional level, a term is presented according to the requirements of the legal *Definition*.

3.4.3 Argumentative Role

An argumentative type of *Definition* is used in law to clarify a controversial term by providing a supporting argument. The Argumentative role of the *Definition* is to provide a reason for believing something is true. It is used to support the classification of legal terms, although each term has its own argumentative weights and reasoning process.

3.5 State of the Art

Research studies are witnessed on information extraction, i.e., Definition Extraction (DE), by using different techniques of Natural Language Processing (NLP), Machine Learning (ML), Deep Learning (DL), Information Retrieval (IR) and Knowledge-Based Extraction (NBE). It is used in various applications, i.e., machine translation, question-answering systems, automatic summarization, ontology engineering, text classification and dictionary construction. Assorted studies have been witnessed on data extraction and definition extraction from legal documents.

Definition extraction is a domain-specific technique that extracts targeted definitions from text or text documents [60][61]. It is a process of identifying and extracting definitions [67]. Studies have witnessed different automated and semi-automated techniques/algorithms that extract Definitions from the text [68][69]. These automated and semi-automated definition extraction techniques significantly reduced the effort and time required in manual extraction. Over the last few years, researchers [70][71][72][73] have worked significantly aiming to achieve DE with varying degrees of success. Definition extraction is becoming a hot topic in Law. In law, definition extraction is performed in detecting and extracting Definitions from Legal Texts (LT) [74][75]. Definitions extraction and annotations were done using semantic relation extraction on English and Slovene texts related to geology, glaciology, and geomorphology [61]. In [94], definition extraction was done on Dutch texts using a combination of structural and linguistic information. The author's

terminology and definitions from mathematical texts were extracted using a rule-based syntactic approach [95]. Automatic definition extraction was done by applying the extraction of candidates for dictionary definitions from unstructured texts in the Serbian language [96]. In [97] a DEFT model was proposed based on NLP for the extraction of definitions from free and semi-structured text. A neural network-based architecture was proposed for the extraction of definitions [98]. In [99] an algorithm was proposed for the definition's extraction from text. Some of the studies on definition extraction are presented in detail below.

A joint model for definition extraction was proposed [100] that was based on semantic consistency and syntactic connection. Three different datasets were used: (i) WCL (World Class Lattices), which consisted of 2847 non-definitional and 1871 definitional sentences that were collected from Wikipedia; (ii) W00, which consisted of 1454 non-definitional and 731 definitional sentences (iii) DEFT, this corpus contained 15339 non-definitional and 5964 definitional sentences. On sentence labelling, the proposed model performs 83.3% in terms of f-score on the WCL dataset, whereas the results on W00 was 60.6%, and DEFT results were 50.6% F-score on Textbook and Contract, the F-score was 71.1%. The experiments were also conducted by consolidating the proposed model with BERT. The consolidated model achieved 85.3% results in terms of F-score on the WCL dataset and 66.9% on the W00 dataset. The model performs 54% and 68.2% F-score on the DEFT corpus on Textbooks and Contracts, respectively.

The ML-based approach was presented [101] for definition extraction. For this purpose, the dataset was mined from Wikipedia and further preprocessed. The dataset, World-Class Lattices (WCL), consisted of 4718 manually annotated sentences. In pre-processing, sentences that were less than five words were tagged as outliers, and sentences with other language words were also removed from the dataset. In the first experiment, Long Short-Term Memory (LSTM) tagging was used by Part of Speech (POS) tagging, and the model achieved a 90.7% F-score at the model's cross-validation. The Trained model has a good understanding of the relation between the words in the sentence. However, when the

results were compared with those of the state of the art, they were not good. The BERT outperforms with a 97.4% F1 score at definition extraction.

A [102] technique for the automatic extraction of legal definitions was proposed. The proposed techniques not only extract the definition of the legal terms but also extract the domain-relevant terms using Data Mining (DM) and Natural Language Processing (NLP) techniques. For this purpose, the dataset that was used was Taiwan's Laws and Regulations database, which was collected from the Ministry of Justice of Taiwan. The data source consists of 585 acts, of which 353 were acts, 216 were social acts, and the remaining 16 were comprehensive acts. Preprocessing of the dataset was performed, and tags were added using Par of Speech (POS) tagging. For this purpose, legal keywords and definitions were gathered, and the ICTCLAS tool was used for the segmentation. The 15 patterns were found for more than one keyword and definition. After extracting patterns and definitions, the ontology constrictions were performed. The total number of patterns extracted classes in the study was 1114, which were compared with the manual extracted legal definitions in which competent authorities' classes were 81. The total number of classes judged by Support Vector Machine (SVM) was 168020, and the total number of classes in law ontology was 169215.

Automatic detection of definitions from legal text is performed [103]. This research was performed to detect the automated way of detecting style checking by error handling. For this purpose, the legislative draft of the Government of Switzerland was issued by the federal administration of the Swiss Confederation. The corpus of legal documents used in this study consists of 1847 Swiss legal and legislative texts, which were downloaded from the Swiss Federal administrative body (www.admin.ch/ch/d/sr/). A total of 1847 legislative texts were selected out of 1915. After extracting the documents that were in HTML format, they were converted into an XML file format and further preprocessed. While preprocessing legislative documents, all the formatting material

was removed distant from line breaks. The representation of special characters was condensed with normal text.

The annotation of chapters, sections, subsections, sub-sub sections, articles, paragraphs, sentences, and lists was performed by pattern-based annotation. To check the results of auto-detection of definition, they annotated 17 legislative texts manually, which were selected from all domains, i.e., 2 of them were selected from constitutional law, 2 of them from education, science and culture law, 2 of them from private law, 2 of them form national defense law, 2 of them form criminal law, 2 of them form finance law, 2 of them from economy law and 3 of them from energy and transport law. After performing manual annotation, the annotation is evaluated by Kappa Statics. Of the total annotated text, 12% of the text was excluded due to the exhibited pattern from the evaluation. The auto-detection of legal definitions on Enumerated Definitions performs 99% precision; on Bracketed Definitions, the precision value is 94%. The authors claim that a substantial number of legal definitions were detected automatically with relatively shallow pattern-matching methods. In [104] a model was designed to assist legal researchers and lawyers access legal data from most applied cases.

Extraction of definitions was performed on Brazilian legal text [105]. The dataset which was used for experimentation in this study is BCTL (Brazilian Collection of Telecommunication Laws). The dataset consists of 1940 reference documents. After the dataset was collected, preprocessing was performed to clean the data from the outliers. After preprocessing the tokenization, the segmentation was performed on paragraphs, sentences, and text using the Punkt algorithm, which was implemented using NLTK, a famous Machine Learning library. As a result, a 6120832 number of tokens corpus was created with words and punctuation. After the segmentation, the part of speech tagging was performed using a Brill tagger. At the evaluation of the tagger, it shows 90.44% results in terms of accuracy. To check the performance of the Definition Extraction, the dataset is split into training data, which is 70% of the total, and the remaining 30% of data is used for testing

purposes. The extractor achieved 73.50% accuracy, whereas the other evaluation measures were also used to evaluate the extractor. The value of precision was 75.60%, the recall was 69.6%, and the value of F-score was 72%.

In [106] a model was proposed for linking the concepts of cross-lingual legal documents from different legal systems to increase their accessibility. Legal document accessibility was supported in different ways, i.e., summarizing content, doing full-text searches, hyperlinking related documents, query-supporting formulation, and using thesaurus-based search. The proposed model was based on linking concepts based on the meaning of the legal concepts and relations among the legal concepts. Two types of relations were distinguished: content and structural concepts. The content relations reflect the differences and similarities between the meanings of the legal concepts, whereas the structural concept reflects the actual connection of legal concepts. Legal violation identification[107] was conducted on unstructured text using large language models. Following the successful detection of legal violations with an F-score of 62.69% using the BERT model, these violations were linked to potentially affected individuals. The victim association achieved a score of 81.02%, demonstrating the effectiveness of the Natural Language Inference (NLI) task performed.

In [108] Lexical Ontologies for Legal Information Sharing (LOIS) was proposed to increase the accessibility of legal documents. The LOIS project entails the development of a large multilingual WordNet for cross-lingual legal information retrieval to ease the accessibility of legal documents for legal professionals. A multilingual WordNet information retrieval is purposed that supports interlingual and monolingual data extraction for automatically extracting Legal *Definition*s from European directives. For the extraction, they used a hybrid approach that was both legal and lexical. The LOIS project consists of 5000 synsets. Table 1 demonstrates the different statistics of the literature review carried out on the extraction of Definition.

Table 1: Literature Review on Definitions Extraction

AUTHOR	DOMAIN	APPROACH	DATASET	DATA SIZE	F/SCORE	ACCURACY	ALGORITHM
Veyseh	Definition Extraction	Semantic & syntactic	1: WCL (World class Lattices) 2: W00 3: DEFT	1: 4718 2: 2185 3: 21303 Sentences	1: 83.3% 2: 60.6% 3: 50.6%	N-A	Self-Proposed & BERT
Kumar	Definition Extraction	Machine Learning	World-Class Lattices (WCL)	4718 Sentences	90.7%	N-A	LSTM
Hwang	Definition Extraction	Patterns Extracting	Taiwan's Laws	585 Acts	N-A	168020 Extracted Patterns	SVM
Höfler	Definition Detection	Pattern Matching	Swiss Legal Acts	1847 Acts	99 Precision	N-A	ML
Ferneda	Definition Extraction	Machine Learning	BCTL (Brazilian Collection of Telecom Laws)	1940 documents	72.0	90.44	Punkt-Algorithm Brill tagger
Mommers	Linking Legal Documents	Pattern Matching	N-A	N-A	N-A	N-A	Hyper-linking
Dini	Definition Extraction	Lexical Ontologies inf-sharing	LOIS Legal Documents	5000 Synsets	1.0	0.271	Word-Net

3.6 Definition Extraction and Annotation

The following methodology is adopted for detecting and annotating Definitions in the EU legislative dataset. The methodology of this study is divided into five phases to perform this task. The first phase involves data acquisition, the second phase involves algorithm design for the detection and annotation of definitions, the results phase presents the results, and in the fourth phase, the verification and validation of results are performed, followed by the discussion of results and the conclusion of the study. The rest of the details are discussed in the following sections of the methodology. The methodology of this research is shown in Figure 5.

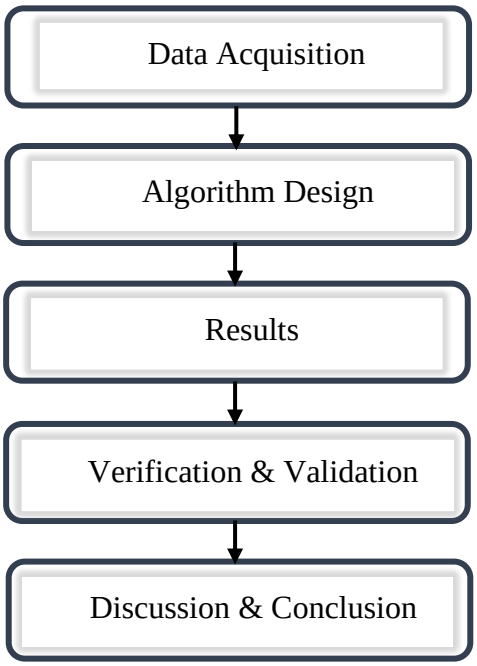

Figure 5: Methodology for the Detection and Annotation of Definitions

3.7 Data Acquisition

The dataset acquired for this study consists of EU legislation. It includes legislative texts related to Agri-Food spanning the period from 1962 to 2021, provided in the Akoma Ntoso (AKN) file format (AKN represents parliamentary, legislative, and judiciary documents in XML format). AKN is an international legal XML standard approved by the OASIS (Organization for the Advancement of Structured Information Standards) body [109]. The total number of legislative documents/files is 15082. Figure 6 shows the dataset detail and number of representations of files in each year.

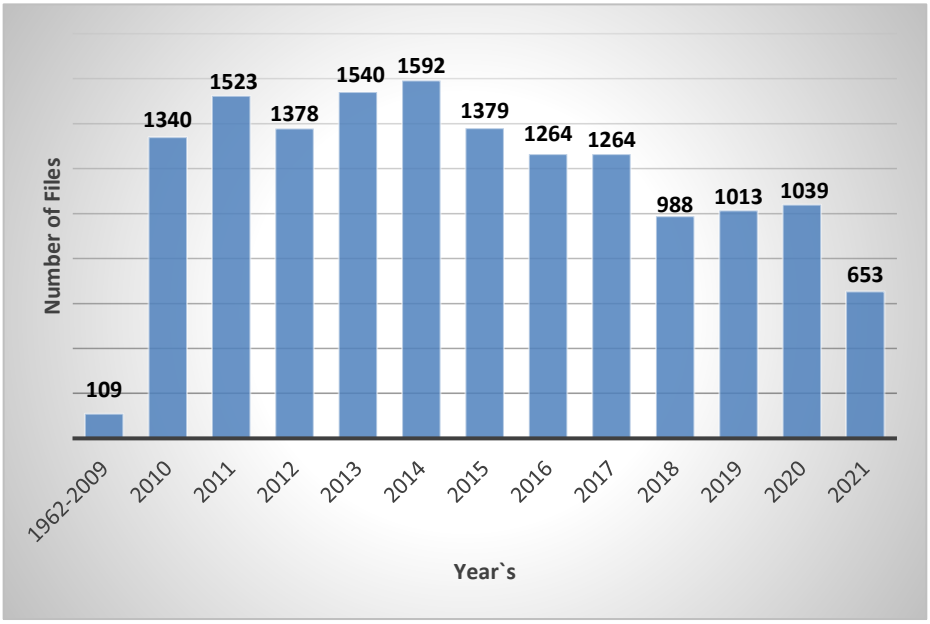

Figure 6: Number of Files per Year

The files are in markup file format. Figure 7 shows a visual representation of a file, including the definition to be annotated.

3.8 Algorithm Design

The annotation is performed on a *Delimiting Definition*s, which is a very famous type of *Definition* used to define norms in legislation. A

Natural Language Processing technique, known as rule-based mining, is used for auto-detection and annotation of legal texts.

```
<?xml version="1.0" encoding="UTF-8"?>
<akomaNtoso xmlns="http://docs.oasis-open.org/legaldocml/ns/akn/3.0" xmlns:fmx="
http://formex.publications.europa.eu/schema/formex-05.56-20160701.xd">
    <act name="regulation/REG">
        <article eId="prt_1__prt_1__art_4" fmx:GUID="004">
            <num>Article 4</num>
            <heading>Definitions</heading>
            <paragraph eId="prt_1__prt_1__art_4__para_1" fmx:GUID="004.001">
                <num>1.</num>
                <list eId="prt_1__prt_1__art_4__para_1__list_1">
                    <intro eId="prt_1__prt_1__art_4__para_1__list_1__intro">
                        <p>For the purposes of this Regulation, the following
                        definitions shall apply:</p>
                    </intro>
                    <point eId="prt_1__prt_1__art_4__para_1__list_1__point_1" defines="
                    #def_1">
                        <num>(1)</num>
                        <content eId="
                        prt_1__prt_1__art_4__para_1__list_1__point_1__content">
                            <p> "credit institution" means an undertaking the business
                            of which is to take deposits or other repayable funds from
                            the public and to grant credits for its own account; </p>
                        </content>
                    </point>
            </paragraph>
        </article>
        </act>
</akomaNtoso>
```

Figure 7: The visual representation of the text in the AKN file

A rule-based system for extracting targeted information from the text is potentially better. It can cope with information better than memory-based systems and easily handle wider and unseen inputs. This study targeted the extraction and annotation of Delimiting *Definition*s from the EU legislative files through the lens of two contrasting scenarios using rule-based mining.

Scenario I:

In scenario I: all the *Delimiting Definition*s will be annotated if they are found under the article's tag and heading tag containing the inner text ‘definition or definitions’. After getting the tag article and tag heading having the word ‘Definition or Definitions’ the *Delimiting* types of *Definition*s are found and annotated.

Scenario II:

In the second scenario, the Independent *Delimiting Definition*s are annotated, which involves extracting and annotating the *Delimiting Definition*s found anywhere within the legislative files.

Further details regarding the extraction and annotation of *Delimiting Definition*s from Scenario-I and Scenario-II are discussed in the next sections.

3.8.1 Scenario-I

For this purpose, the Element Tree (ET) library, of Python, is used for creating a structured tree to handle AKN files easily. Experimental Details are presented in Table 2. In legal documents, formatted in a markup language, (AKN format), first, must find the heading tag (τ_H) having Targeted keywords, 'Definition' or 'Definitions' ($k_{Definition}$or $k_{Definitions}$). After finding the keyword enter τ_H and find the paragraph tag (τ_ρ). After finding τ_ρ enter body of τ_ρ and find the keyword "means" (k_{means}) being followed by some quoted text. Store this quoted text ($Q_{Definiendum}$). Annotate $Q_{Definiendum}$ as per the rules specified in $Equation\ 1$, and then replaced with new quoted text ($Q'_{Definiendum}$).

$$Q'_{Definiendum} = <\text{def}\ \ eId = \text{def}\ \_i > Q_{Definiendum} </def> \quad (1)$$

Next, store the rest of the text in the body of τ_ρ i.e., T_{Next}. Annotate T_{Next} as per the rules defined in $Equation\ 2$ and then replace T_{Next} with T'_{Next}.

$$T'_{Next} = < defBody\ eId = "defBody_{\_i} > T_{Next} </defBody> \quad (2)$$

Where 'i' is the unique numeric identifier assigned in this annotation, and it is the same in both equations.

After detecting the equations using rule-based mining, the annotation is done as explained above in $Equation\ 1$ and $Equation\ 2$. The Definiendum is tagged as the heading of *Definition*. The Definiendum is

marked with a start tag i.e. "$< def$ ... >" and an end tag i.e. "$</def >$". Rest, i.e. the text that follows "means" and ends at either "," or ";" is marked as the body of the *Definition* that is annotated with the start tag "$< defBody$ ... >" and end tag "$</defBody >$". All siblings and childrens are annotated in the same way. After the annotation of the Definiendum (def) and Rest (defBody), a unique identification number (eId`s) is assigned to all the annotated Definiendum and defBodies. All the annotated material (outputs) are saved using the Pretty XML library. The camelCase [110] is used to avoid inconsistencies in format. The indentation method is used to validate AKN files.

For example, let's have the following short text [111] and visual representation of this process is shown in Figure 8.

> In the $< p >$ tag where the quoted text, a Definiendum, appears, the $< def >$ tag is added to highlight the definition heading, e.g., $< p >$ " $< def\ eId = "def\_1" >$ pyrotechnic article $</def >$ ". The word 'means' indicates the beginning of the definition body, which is annotated as $< defBody\ eId = "defBody\_1" >$, marking the content as the body of the definition; $</defBody >$ is used to highlight the end. CamelCase is enforced to maintain consistency in the AKN file format. Additionally, a unique ID (eId) is added to each heading and body of the definition for differentiation.

The detection and annotation of norms in AKN files that fall under the tag of definition are extracted and annotated in two steps: first, the annotation rules are fine-tuned, and second, the algorithm is designed for the detection and annotation of Norms in EU Legislation.

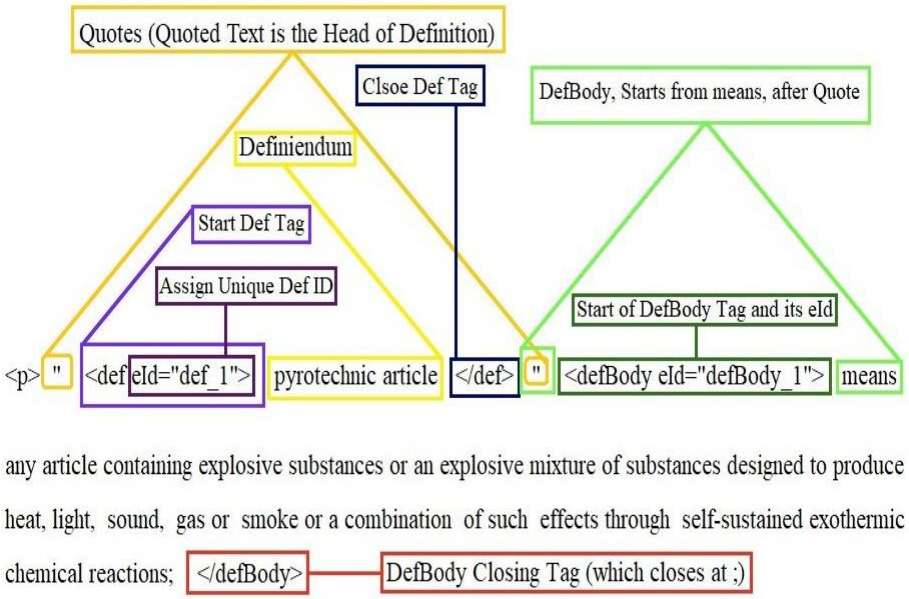

Figure 8: Visual representation on how to perform annotation

3.8.1.1 Basic Idea

The first task is performing annotation on the *Definition* is to detect the targeted text. This research follows some rules for the detection of targeted text. First, to find the articles tag and then find the heading tag in the file containing the inner text ‘Definition or Definitions’. If the heading tag is found with the content ‘Definition or Definitions’, then enter it into the heading tag, and the next task is to find the quoted text, followed by the word ‘means’. If the quoted text is found followed by the word ‘means’, then the annotation is performed.

3.8.1.2 Algorithm

An algorithm is designed based on the rules to detect *Delimiting Definition*s under the heading tag with the inner text ‘Definition or Definitions’ under the articles' tag in legislative files to find an exact *Definition*, *Definition* body and subparts. To detect the *Definition* from the XML file, rule-based (a technique that uses predefined rules to extract Targeted text from the dataset) mining is used to find the exact targeted text of the

Definiendum (Head of *Definition*), the body of the *Definition* and the subparts of the *Definition* body. After detecting the targeted material, all the material, i.e., *Definition* (def), *Definition body* (defBody) and their subparts, are annotated. The algorithm for detecting and annotating *Delimiting Definition*s from legislative documents is present in AKN. The pseudocode (shown in Figure 9) is used to annotate the *Definition*s.

```
//all definitions and tags that will be generated and stored in the file will also be stored in a linked list so that
 all information can be stored in a tag i.e., analysis at the end of the file.
Create TagsList
If (fileAsPerRules(file, rules) = True)
   While (!endOfFile())
      If (findArticleTag() = True) ;
         enter_into_article_tag();
         If (findDefinitionTag() = True) ;                          //1
            enter_into_definition_tag();
            Term = findQuatedText () ;                              //2
            If (Term != NULL)
               DefBody_text = find_defBody_text_after_means()       //3
               If (DefBody_text!= NULL)
                  uniqueEID=generateUniqueEID ()                    //4
                  AddDefTag (uniqueEID,Term)                        //5
                  AddDefBodyTag (uniqueEID,Def_body_text)           //6
                  Count_sub_list=0
                  If (sub_list_exist()=true)
                     While (! endOfList_Tag())                       //7
                        LocalListId=generateLocalListId(UniqueEID)//8
                        Count_sub_list=count_sub_list+1
                  TagsList[i].add(uniqueEID, Term,count_sub_lists)
AddMetadataToAnalysisTag();                                         //9
```

Figure 9: The Pseudocode for Annotation of Definition for Scenario-I

The explanation of all the given numbers in the pseudocode is given below in detail:

//1 this function would return true if a tag titled ‘Definition or Definitions’ is found or contains text

//2 this function will search and return a Term/string that is in double quotes, this is a loop that will continue its search until it finds the next article

//3 this function finds the word means and then returns all the text next to it

//4 e.g. 1

//5 <def eId="def_1"> Term going to be defined </def>
//6 <defBody eId=defBody_1> def body text </defBpdy>
//7 this is search </List> and will terminate this loop
//8 add <defBody eId=defBody_1-1> def body text </defBody> the list of defBody>
//9 add all TagsList items in analysis tag with proper formatting (analysis source="#cirsfid", definitions source="#unibo", definition refersTo="#term", definitionHead href="#def_1" refersTo="#term", definitionBody href="#defBody_1" and definitionBody href="#defBody_1-1") if list contains no data, then print "File not following the rules so cannot be annotated".

3.8.2 Scenario-II

In detecting and annotating independent *Delimiting Definition*s in AKN, this study will extract and annotate the *Delimiting Definition*s in AKN in the same way as in scenario I. The only difference is that this study only extracted and annotated the independent *Delimiting Definition*s in AKN in the second scenario, which means that all the *Delimiting Definition*s present in the documents are annotated. As explained in the algorithm for scenario II, where the Quoted text is found followed by the word means, it will be annotated as presented in $Equation\ 1$ and $Equation\ 2$. The rest of the things will be annotated the same way in the first scenario.

3.8.2.1 Basic Idea

The targeted text in the legislative files for annotating independent *Delimiting Definition*s for scenario II is ‘quoted text followed by the word ‘means’’ wherever in the legislative file. If the quoted text is found followed by the word ‘means’ wherever in the legislative files, then the annotation is performed.

3.8.2.2 Algorithm

To detect and annotate the independent *Delimiting Definition*s in legislation, the below algorithm shown in Figure 10 is defined, which

successfully detects and annotates the *Definition*, body and their sub-parts.

```
//all definitions and tags that will be generated and stored in the file will also be stored in a linked list so
 that all information can be stored in a tag i.e., analysis at the end of the file.
Create TagsList
If (fileAsPerRules(file, rules) = True)
    While (!endOfFile())
       Term = findQuatedText () ;
            If (Term != NULL)
               DefBody_text = find_defBody_text_after_means()     //3
               If (DefBody_text!= NULL)
                  uniqueEID=generateUniqueEID ()                  //4
                  AddDefTag (uniqueEID,Term)                      //5
                  AddDefBodyTag (uniqueEID,Def_body_text)         //6
                  Count_sub_list=0
                  If (sub_list_exist()=true)
                     While (! endOfList_Tag())                     //7
                        LocalListId=generateLocalListId(UniqueEID)//8
                        Count_sub_list=count_sub_list+1
                  TagsList[i].add(uniqueEID, Term,count_sub_lists)
AddMetadataToAnalysisTag();                                       //9
```

Figure 10: The Pseudocode for annotation of Definition for Scenario-II

The number explanation is the same as given in the above Pseudocode for scenario 1.

3.9 Results

In this research, the AKN-converted EU legislation files are annotated. First, the targeted text (according to both scenarios) is searched from these files using algorithms designed in the last phase. Once the required text is found, the text is updated according to *Equation* 1 and *Equation* 2.

3.9.1 Experimental Setup

The system with an i7 processor and 16 GB RAM is used for experimentation. The operating system used is Windows 11. The programming

language used in this research is Python 3.11. The framework used for running Python programs is PyCharm 2023.1. The libraries used are Element Tree, Pathlib, Minidom, Etree, and PrettyXML libraries. The 'camelCase' is used for the naming convention, and the indentation method is used to validate AKN files. The same machine is used to classify SDGs and to link the EU legislative files with SDGs at the Goals and Targets level. Experimental-setup details summarized in Table 2.

3.9.2 Scenario-I

The Data Acquisition (see section 3.7) outlines that the dataset comprises 15,082 AKN files. Out of these, 899 files contained the necessary material (*Delimiting Definition*s under the articles tag in the heading tag having the inner text 'definition or definitions') and were successfully annotated in Scenario I. The remaining 14,183 AKN files don't have the annotated material as per the rules defined in scenario-I were left as same they are present. This was due to the absence of required tags in these files despite their standard format, leading to their exclusion from further processing.

3.9.3 Scenario-II

As mentioned in scenario II, the independent *Delimiting Definition*s are extracted and annotated in AKN. The 1272 files have found independent *Delimiting Definition*s in them, and these files are annotated successfully. The remaining 13810 AKN files did not have contents (*Delimiting Definition*s) to be annotated, so these files were skipped for further processing.

By following the rule, all the Definitions and the body of the Definitions are annotated as per define in $Equation\ 1$ and $Equation\ 2$. In the annotation of Definitions, a unique identification number (eId) is assigned to each heading of the definition and the body of the definitions, which is incremented when the next definition is found in the file. If any file has sub- and sub-subparts, they are also annotated. Figure 11 shows a sample text before the annotation process and has the required tags for

the annotation. Figure 12 shows the outcome of the process explained above.

Table 2: Experimental Setup Details

Module	**Specifications**
Operating System	Windows 11
RAM	16GB
Processor	i7
Framework	PyCharm 2023.1
Programming Language	Python 3.11
Libraries	Element Tree
	Pathlib
	Minidom
	spaCy
	Data pretty printer
	Regular Expression Operation
	HTML
	Etree
	PrettyXML
Naming Convention	camelCase
Validation of AKN	Indentation

```
<num>Article 2</num>
<heading>Definitions</heading>
<list eId="art_2__list_1">
  <intro eId="art_2__list_1__intro">
    <p>In addition to the definitions set out in Article 2 of Directive 2010/30/EU, the
    following definitions shall apply for the purposes of this Regulation:</p>
  </intro>
  <point eId="art_2__list_1__point_1">
    <num>(1)</num>
    <intro eId="art_2__list_1__point_1__intro">
      <p>"water heater" means a device that:</p>
    </intro>
    <list eId="art_2__list_1__point_1__list_1">
      <point eId="art_2__list_1__point_a__list_1__point_a">
        <num>(a)</num>
        <content eId="art_2__list_1__point_a__list_1__point_a__content">
          <p>is connected to an external supply of drinking or sanitary water;</p>
        </content>
      </point>
      <point eId="art_2__list_1__point_1__list_1__point_b">
        <num>(b)</num>
        <content eId="art_2__list_1__point_1__list_1__point_b__content">
          <p>generates and transfers heat to deliver drinking or sanitary hot water at
          given temperature levels, quantities and flow rates during given intervals;
          and</p>
        </content>
      </point>
      <point eId="art_2__list_1__point_1__list_1__point_c">
        <num>(c)</num>
        <content eId="art_2__list_1__point_1__list_1__point_c__content">
          <p>is equipped with one or more heat generators;</p>
        </content>
      </point>
    </list>
  </point>
  <point eId="art_2__list_1__point_2">
    <num>(2)</num>
    <intro eId="art_2__list_1__point_2__intro">
      <p>"heat generator" means the part of a water heater that generates the heat
      using one or more of the following processes:</p>
    </intro>
    <list eId="art_2__list_1__point_2__list_1">
      <point eId="art_2__list_1__point_2__list_1__point_a">
        <num>(a)</num>
        <content eId="art_2__list_1__point_2__list_1__point_a__content">
          <p>combustion of fossil fuels and/or biomass fuels;</p>
        </content>
      </point>
      <point eId="art_2__list_1__point_b__list_1__point_b">
        <num>(b)</num>
        <content eId="art_2__list_1__point_b__list_1__point_b__content">
          <p>use of the Joule effect in electric resistance heating elements;</p>
        </content>
      </point>
      <point eId="art_2__list_1__point_2__list_1__point_c">
        <num>(c)</num>
        <content eId="art_2__list_1__point_2__list_1__point_c__content">
          <p>capture of ambient heat from an air source, water source or ground source,
          and/or waste heat;</p>
        </content>
      </point>
    </list>
  </point>
```

Figure 11: Content before Annotation

```
<num>Article 2</num>
<heading>Definitions</heading>
<list eId="art_2__list_1">
    <intro eId="art_2__list_1__intro">
        <p>In addition to the definitions set out in Article 2 of Directive 2010
        /30/EU, the following definitions shall apply for the purposes of this
        Regulation:</p>
    </intro>
    <point eId="art_2__list_1__point_1" defines="#def_1">
        <num>(1)</num>
        <intro eId="art_2__list_1__point_1__intro">
            <p> "<def eId="def_1">water heater</def>" <defBody eId="defBody_1">
            means a device that:</defBody> </p>
        </intro>
        <list eId="art_2__list_1__point_1__list_1">
            <point eId="art_2__list_1__point_a__list_1__point_a">
                <num>(a)</num>
                <content eId="art_2__list_1__point_a__list_1__point_a__content">
                    <p> <defBody eId="defBody_1-1">is connected to an external
                    supply of drinking or sanitary water;</defBody> </p>
                </content>
            </point>
            <point eId="art_2__list_1__point_1__list_1__point_b">
                <num>(b)</num>
                <content eId="art_2__list_1__point_1__list_1__point_b__content">
                    <p> <defBody eId="defBody_1-2">generates and transfers heat
                    to deliver drinking or sanitary hot water at given
                    temperature levels, quantities and flow rates during given
                    intervals; and</defBody> </p>
                </content>
            </point>
            <point eId="art_2__list_1__point_1__list_1__point_c">
                <num>(c)</num>
                <content eId="art_2__list_1__point_1__list_1__point_c__content">
                    <p> <defBody eId="defBody_1-3">is equipped with one or more
                    heat generators;</defBody> </p>
                </content>
            </point>
        </list>
    </point>
    <point eId="art_2__list_1__point_2" defines="#def_2">
        <num>(2)</num>
        <intro eId="art_2__list_1__point_2__intro">
            <p> "<def eId="def_2">heat generator</def>"
                <defBody eId="defBody_2"> means the part of a water heater that
                generates the heat using one or more of the following processes:
                </defBody> </p>
        </intro>
        <list eId="art_2__list_1__point_2__list_1">
            <point eId="art_2__list_1__point_2__list_1__point_a">
                <num>(a)</num>
                <content eId="art_2__list_1__point_2__list_1__point_a__content">
                    <p> <defBody eId="defBody_2-1">combustion of fossil fuels
                    and/or biomass fuels;</defBody> </p>
                </content>
            </point>
            <point eId="art_2__list_1__point_b__list_1__point_b">
                <num>(b)</num>
                <content eId="art_2__list_1__point_b__list_1__point_b__content">
                    <p> <defBody eId="defBody_2-2">use of the Joule effect in
                    electric resistance heating elements;</defBody> </p>
                </content>
            </point>
            <point eId="art_2__list_1__point_2__list_1__point_c">
                <num>(c)</num>
                <content eId="art_2__list_1__point_2__list_1__point_c__content">
                    <p> <defBody eId="defBody_2-3">capture of ambient heat from
                    an air source, water source or ground source, and/or waste
                    heat;</defBody> </p>
                </content>
            </point>
        </list>
</point>
```

Figure 12: The Sample Annotated Definitions

After performing the successful annotation, all the annotated information is added at the end of the active modification tag in each AKN file. Figure 13 shows a sample of this annotation.


```
    <definitions source="#unibo">
        <definition refersTo="#waterHeater">
            <definitionHead href="#def_1" refersTo="#waterHeater"/>
            <definitionBody href="#defBody_1"/>
            <definitionBody href="#defBody_1-1"/>
            <definitionBody href="#defBody_1-2"/>
            <definitionBody href="#defBody_1-3"/>
        </definition>
        <definition refersTo="#heatGenerator">
            <definitionHead href="#def_2" refersTo="#heatGenerator"/>
            <definitionBody href="#defBody_2"/>
            <definitionBody href="#defBody_2-1"/>
            <definitionBody href="#defBody_2-2"/>
            <definitionBody href="#defBody_2-3"/>
        </definition>
        <definition refersTo="#ratedHeatOutput">
            <definitionHead href="#def_3" refersTo="#ratedHeatOutput"/>
            <definitionBody href="#defBody_3"/>
        </definition>
        <definition refersTo="#standardRatingConditions">
            <definitionHead href="#def_4" refersTo="#standardRatingConditions"/>
            <definitionBody href="#defBody_4"/>
        </definition>
        <definition refersTo="#biomass">
            <definitionHead href="#def_5" refersTo="#biomass"/>
            <definitionBody href="#defBody_5"/>
        </definition>
        <definition refersTo="#biomassFuel">
            <definitionHead href="#def_6" refersTo="#biomassFuel"/>
            <definitionBody href="#defBody_6"/>
        </definition>
        <definition refersTo="#fossilFuel">
            <definitionHead href="#def_7" refersTo="#fossilFuel"/>
            <definitionBody href="#defBody_7"/>
        </definition>
        <definition refersTo="#hotWaterStorageTank">
            <definitionHead href="#def_8" refersTo="#hotWaterStorageTank"/>
            <definitionBody href="#defBody_8"/>
        </definition>
        <definition refersTo="#back-upImmersionHeater">
            <definitionHead href="#def_9" refersTo="#back-upImmersionHeater"/>
            <definitionBody href="#defBody_9"/>
        </definition>
        <definition refersTo="#solarDevice">
            <definitionHead href="#def_10" refersTo="#solarDevice"/>
            <definitionBody href="#defBody_10"/>
        </definition>
        <definition refersTo="#solar-onlySystem">
            <definitionHead href="#def_11" refersTo="#solar-onlySystem"/>
            <definitionBody href="#defBody_11"/>
        </definition>
        <definition refersTo="#packageOfWaterHeaterAndSolarDevice">
            <definitionHead href="#def_12" refersTo="
            #packageOfWaterHeaterAndSolarDevice"/>
            <definitionBody href="#defBody_12"/>
        </definition>
        <definition refersTo="#heatPumpWaterHeater">
            <definitionHead href="#def_13" refersTo="#heatPumpWaterHeater"/>
            <definitionBody href="#defBody_13"/>
        </definition>
    </definitions>
</analysis>
```

Figure 13: The Annotated Information

3.10 Verification and Validation

Annotation of data is a static process in which explanatory remarks are added to data. This technique provides additional information, interpretation, and clarification regarding the text. Annotations can be conducted manually by reviewing documents one by one or through automated or semi-automated tools. In the fields of AI and law, annotation plays a crucial role, particularly in the legal and compliance analysis of legislation. In this context, annotation provides explanatory remarks, comments, notes, or interpretations of legal texts. The goal of annotating legislation is to elucidate legislative clauses and clarify their meanings to help readers understand the legal implications. Various forms of legislative annotation exist, including definition annotation, legislative intent, case references, and cross-references. This study specifically focuses on annotating definitions using rule-based mining techniques within Natural Language Processing (NLP). The verification and validation of the performed annotation of *Delimiting Definition*s is discussed in detail in this section. To verify the results of auto-annotation of *Delimiting* type of *Definition*s, the manual annotation is performed and then the results of auto-annotation is compared with the manual annotation. The rest of the details of manual annotation are given below.

3.10.1 Manual Annotation

Manual annotation is considered the most accurate annotation [112]. It is a way to annotate the *Definition*, one by one, by reading manually. But with a large dataset, it becomes time-consuming and humanly impossible even with additional human and other resources [113]. These limitations lead to an automated solution to the problem. In this research, auto-annotation is being used to *Delimiting* the type of *Definition*s of EU legislation. To validate the results of the auto-annotation of the *Definition*s, manual annotation of the *Definition* is also performed.

Manual annotation of *Definition*s is performed in four phases [47] [114]. Details are as follows:

3.10.1.1 Guidelines Preparation and Annotator Selection

In this phase, the selection of annotators and complete guidelines are prepared to annotate the *Delimiting Definition*s.

3.10.1.1.1 Annotator Selection

The most important thing for manual annotation is the selection of the annotators. For this purpose, three annotators were selected. The volunteers were well-educated and were familiar with the *Definition*s in legislation with the concepts of *Definition*s annotation and had a good grip on the law [47].

3.10.1.1.2 Guidelines Preparations

In this phase, complete guidelines are prepared to annotate the *Delimiting Definition*s. The following guidelines were prepared and provided to annotators for the manual annotations.

- G1: Open the AKN file in 'Sublime Text'
- G2: Find tag, *Article, i.e. <num>Article 4</num>*
- G3: Find tag, *Heading, i.e., <heading>*
- G4: Find keywords, *Definition/Definitions in Heading i.e., <heading>Definitions</heading>*
- G5: Find tag, *P i.e, <p>*
- G6: Find quoted text in the tag *P* that is followed by "means," i.e., "credit institution" means an undertaking….
- G7: Add new tag, *def with eId i.e., "<def eId="def_1">credit institution</def>"*
- G7: Add new tag, *defBody, with the same eId in the def i.e., <defBody eId="defBody_1">*

The above-mentioned seven guidelines will be followed for scenario I. Whereas for scenario II, guidelines 5 to 7 will be followed.

3.10.1.2 Phase II-Annotators' Training

In this phase, the services of three volunteers—V1, V2 and V3, are acquired to annotate the *Delimiting Definition*s from EU legislation. Initially, they were introduced to the annotation process. Then, a hands-on training session on annotation guidelines was conducted. During the session, their issues and conflicts are resolved.

3.10.1.3 Phase III-Conflict Resolution

In this phase, the dataset (files having *Definition*s to be annotated) was given to the annotators for annotation. For conflict resolutions, a file of EU legislation is given to the first two annotators—annotator V1 and annotator V2. Once they had completed the annotation, a short meeting was arranged to resolve the conflicts by involving the third annotator too. After the conflict resolution, the same file is given to the 3rd annotator, V3, for annotation of the *Definition*s in the file for practice.

3.10.1.4 Phase IV-Computation of Inter-Annotator Agreement

After the training and conflict resolution, the 12 same files were given to all three annotators for annotation. After performing the manual annotation, these annotated files are compared, and the inter-annotator agreement is computed using Kappa Statics. The formula to compute the Kappa coefficient is shown in $Equation\ 3$. The p_o is the observed agreement between the annotators and p_e is the expected agreement of random judgements.

$$k = ((p_o - p_e))/((1 - p_e)) \quad (3)$$

According to Cohen`s Kappa [115] [116], if the agreement between the annotators is '0', it shows no agreement has been reached between

the annotators. The annotators agree slightly if the score is between 0.10-0.20. The value of Cohens is 0.21-.40 shows a fair agreement between annotators is found. There is moderate agreement when the value is between 0.41 and 0.60. When the value of Cohen`s is found between 0.61-0.80, substantial agreement is found, and 0.81-0.99 value shows a nearly perfect agreement between the annotators. The value 1 is for perfect agreement. Table 3 summarizes the interpretation of Cohen`s Kappa on different values [117].

Table 3: Interpretation of Cohen`s Kappa

Cohen`s Kappa values	**Interpretations**
0	No Agreement
0.10-0.20	Slight Agreement
0.21-0.40	Fair Agreement
0.41-0.60	Moderate Agreement
0.61-0.80	Substantial Agreement
0.81-0.99	Near Perfect Agreement
1	Perfect Agreement

A total of 394 *Definition*s were to be annotated across the 12 selected files, with one file chosen from each year. Annotator 1 successfully annotated 383 *Definition*s, missing one *Definition* from the second file, two from the third file, two from the fourth file, two from the fifth file, one from the sixth file, one from the seventh file, one from the eighth file, and also one from the tenth file. The annotator V1 successfully annotated the rest of the *Definition*s of the eleventh and twelfth files.

The second annotator also annotated 381 *Definition*s, missing two *Definition*s from the third file: one from the fifth, one from the seventh, two from the ninth, one from the tenth, one from the eleventh, and one from the twelfth. Annotator 3 annotated 385 *Definition*s but missed one *Definition* from the fourth file, one from the sixth, two from the eighth, three from the ninth, one from the eleventh, and one *Definition* from the

twelfth file. A comparison of the performance among different annotators and the auto-annotation process is presented in Table 4.

Table 4: Results of the manual annotation showing the number of *Definition*s annotated per file for all three volunteer annotators (V1, V2, and V3), as well as the best manual annotation among the three annotators, are presented

File #	Number of Definitions added by each annotator			Best Manual Annotation
	V1	V2	V3	
32010L0063	8	8	8	8
32011R0458	17	18	18	18
32012R0528	31	31	33	33
32013L0030	33	35	34	35
32014L0090	20	21	22	22
32015R1186	26	27	26	27
32016L0797	40	38	41	41
32017R0746	65	66	64	66
32018R0858	57	54	54	57
32019L0882	42	41	43	43
32020R1503	18	17	17	18
32021R0817	26	25	25	26

Based on these statistics of different annotators' performance, their inter-annotator agreement, i.e., the Kappa Coefficient, is calculated using $Equation$ 3. The value of inter-annotator agreement between all three annotators is 0.972%; calculations are shown in $Equation$ 4. This is "Near Perfect Agreement," which is reliably excellent using Cohen's table shown in Table 4.

$$k = \frac{(394*383 - 394)}{(394*394 - 394)} = 0.972 \quad (4)$$

After reviewing the manual annotation and computing their inter-annotator agreement, the best-annotated files are separated to compare with auto-annotated files. Table 5 shows the comparison of the best manual annotation and auto-annotation. The auto-annotation of the Definition is compared with manual annotation, and the computational value of the annotation is calculated using Kappa Statistics.

Table 5: Comparison of Auto and Manual Annotation

File #	Best Manual Annotation	Auto-Annotation
32010L0063	8	8
32011R0458	18	18
32012R0528	33	33
32013L0030	35	35
32014L0090	22	22
32015R1186	27	27
32016L0797	41	41
32017R0746	66	66
32018R0858	57	57
32019L0882	43	43
32020R1503	18	18
32021R0817	26	26

3.11 Verification of Annotation

To compute the annotation results, the auto-annotation was compared with the best manual annotated files and the computational value was calculated using Cohen`s Kappa coefficient. The count of best manual annotated files was 394 from 12 selected files, whereas, in the results of auto annotation, the number of annotated *Definition* was also 394, which means that the value of the Kappa coefficient is 1; calculations are shown in $Equation$ 5.

$$k = \frac{(394*394 - 394)}{(394*394 - 394)} = 1 \quad (5)$$

The Calculated value of the Kappa coefficient is 1 between the best manual annotation and the auto-annotation, indicating that the algorithm devised for performing auto-annotation is very much promising.

3.12 Comparison of Annotation

To compute the annotation results, the auto-annotation is compared with the best manual annotated files and the computational value is calculated using Cohen`s Kappa coefficient. The count of best manual annotated files is 394 from 12 selected files, whereas, in the results of auto annotation, the number of annotated *Definition*s is also 394, which means that the value of the Kappa coefficient is 1, the calculation is shown in $Equation\ 5$.

$$k = \frac{(394*394 - 394)}{(394*394 - 394)} = 1 \quad (6)$$

The Calculated value of the Kappa coefficient is 1 between the best manual annotation and auto-annotation, which means that the performed annotation is perfect.

3.13 Application of Definitions Annotation

The EU and FAO of the UNO are working collaboratively to address global hunger issues by preserving food. The EU formulates policies concerning food safety. The annotation of definitions necessitates the alignment of these policies to evaluate their effectiveness against the SDGs framework [118]. These annotations can also be used in different applications. Some of the applications of the Definitions' Annotations are listed below:

It can be used to:

i. Annotation of definitions is helpful in the fine-tuning of LLM (Large Language Model) in the legal domain.
ii. Annotated definitions can be used to classify the EU

legislations based on their alignment with the United Nations Sustainable Development Goals.

iii. Classify the legislative text, topic modelling, or keyword extraction.

iv. Assess the focus of European Union policies with the Sustainable Development Goals.

v. Study the integration of sustainability in the legal frameworks.

vi. Extract entities, relationships, and concepts to build legal knowledge graphs aligned with SDGs.

vii. Track the legislative focus of the EU over time concerning specific SDGs.

viii. Compare how different EU member states or regions address SDGs in national transpositions of EU directives.

ix. Correlate legal interventions with sustainability indicators to evaluate policy effectiveness.

x. Train systems to infer legal obligations, rights, and prohibitions based on annotated text.

xi. The annotated dataset can be used to build tools that check policy drafts or business processes against SDG-aligned legal requirements.

xii. Identify argument structures in legal texts related to sustainability topics.

xiii. Generate citizen-friendly summaries or suggestions for policy improvement based on SDG alignment.

xiv. Aid policymakers in drafting SDG-consistent legislation using precedent-based recommendations.

xv. Develop dashboards showing which legislative areas are under-addressed relative to sustainability targets.

3.14 Discussion

In this chapter, the detection and annotation of *Definition*s are performed on the *Delimiting Definition*s through two consecutive scenarios, which are discussed in detail (see section 3.8.1 and section 3.8.2) above, using the EU legislation (which consists of EU legislation from the span

of time 2010 to 2024). In Scenario I, 899 files contain annotated material, and these files are successfully annotated according to the rules defined in the algorithms (see section 3.8.1.2). In the second scenario, 1,272 files with annotated material were processed as per the rules defined in algorithms (see section 3.8.2.2). The remaining files did not fulfil the basic requirements of annotations and did not find any annotated material, so those were unchanged. The auto-annotations process, i.e. algorithms, are verified and validated through the standard process using Kappa statistics (discussed in section 3.11). The statistics of the annotations of *Delimiting Definition*s and NER are presented in Table 6.

Table 6: Statistics of definitions annotations

Properties	**Number**
Files Annotated in Scenario I	899
Files Annotated in Scenario II	1272
Total number of definitions Annotated	11705
Definitions found under the definition tag	10617
Definitions Found in Recitals	32
Definitions Found in Article 1	1018
Definitions Found in Article 2	5817
Definitions Found in Article 3	2849
Definitions Found in Article 4	959
Definitions Found in Article 5	136
Number of files containing NER <location>	135
Number of files containing NER <time>	42
Number of files containing NER <date>	221
Total NER-tagged <locations>	381
Total NER-tagged <time>	56
Total NER-tagged <date>	497
Number of files having No NER	933

The results of this study provide valuable insights into the annotation of *Delimiting Definition*s. The results found that 90.7% of *Definition*s are under the article tag in the heading tag containing the word 'definition or definitions', while the remaining 9.3% are distributed throughout other file sections. In total, 1018 *Definition*s were identified in Article 1, representing 8.7% of the overall annotated *Definition*s. Article 2 contains 49.69% of the total *Definitions*, a count of 5,817 *Definition*s. In Article 3, 2849 *Definitions* were identified, accounting for 24.34% of the total *Definition*s. The 8.19% of total *Definition*s are present in Article 4, the 959 total. Article 5 includes approximately 1.16% of the total *Definition*s, with 136 *Definition*s in total. During the analysis, 32 *Definition*s were found under the Recitals tag across 32 files, each containing one *Definition*s in the recitals. Furthermore, 1018 *Definition*s, representing 9.29% of the total, were not categorized under the articles' tag in the heading tag having the inner text 'Definition or Definitions' within the legislative text. The visual representation of found the number of definitions in first five articles is shown in Figure 14.

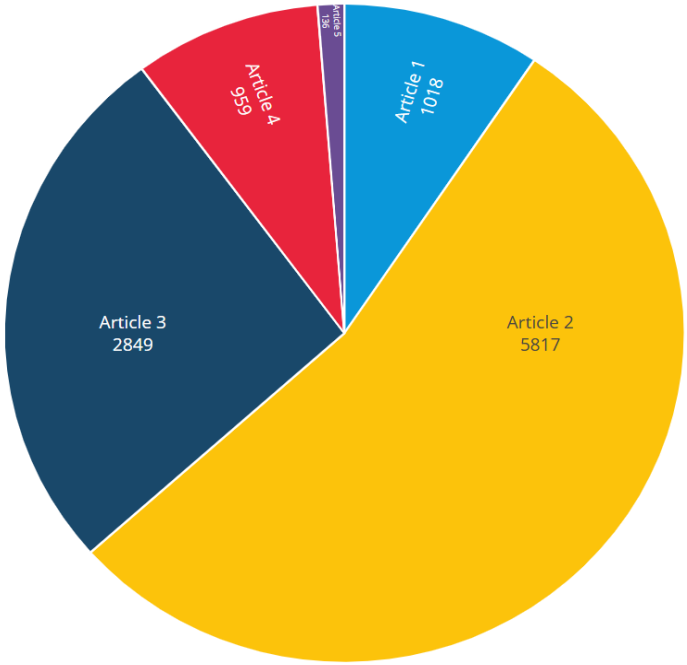

Figure 14: Definitions Representation Found in First Five Articles

After successfully detecting and annotating the *Delimiting Definition*s in EU legislative files, Named Entity Recognition (NER) was implemented using the spaCy library. This is a widely used NLP tool for identifying named entities in unstructured text and categorizing them into standard classifications such as locations, time expressions, person names, organizations, quantities, and monetary values.

NER was applied for recognizing GPE (Geopolitical Entity), Date, and Time, which were then successfully annotated GPE as $< location > Italy </location >$, Date as $< date > April\ 2020 </date >$, and Time $< time > 1600UTC </time >$. For NER, this study tried different language models, the Small Model (sm), Medium Model (md), and the Large Model (lg). This study also tries the Transformer (trf) model, a large and more powerful model based on BERT (Bidirectional Encoder Representations from Transformers) and RoBERTa. However, the results of the transformer model were found to be more satisfying, so this study uses the transformer model for the NER in legislative definition. Visualization of NER tagging is shown in Figure 15.

<def eId="def_75">"residential property"</def> <defBody eId="defBody_75"> means a residence which is occupied by the owner or the lessee of the residence, including the right to inhabit an apartment in housing cooperatives located in <location>Sweden</location>, the minimum period of notice for a voluntary exit of a subsidiary from the liability arrangement is <date>10 years</date> continuous period of <time>24 hours</time>.

Figure 15: NER Tagging Representation

Among the 1272 legislative files, 398 had GPEs, and 933 did not have GPEs in the annotation Definition. The GPE 'location' is found in 135 files and occurs 381 times. The GPE time is found 56 'times' in 42 files, and the GPE 'data' occurrence is 497 times in 221 files. All the GPEs found in the annotated Definition were tagged successfully.

Based on the statistics above, this study concluded that Definitions are found under the Definitions heading and in the recitals. Therefore, this study proposed that all Definitions should be clearly stated in the legislation under the definitions heading. The tagging of NER is shown in Figure 16.

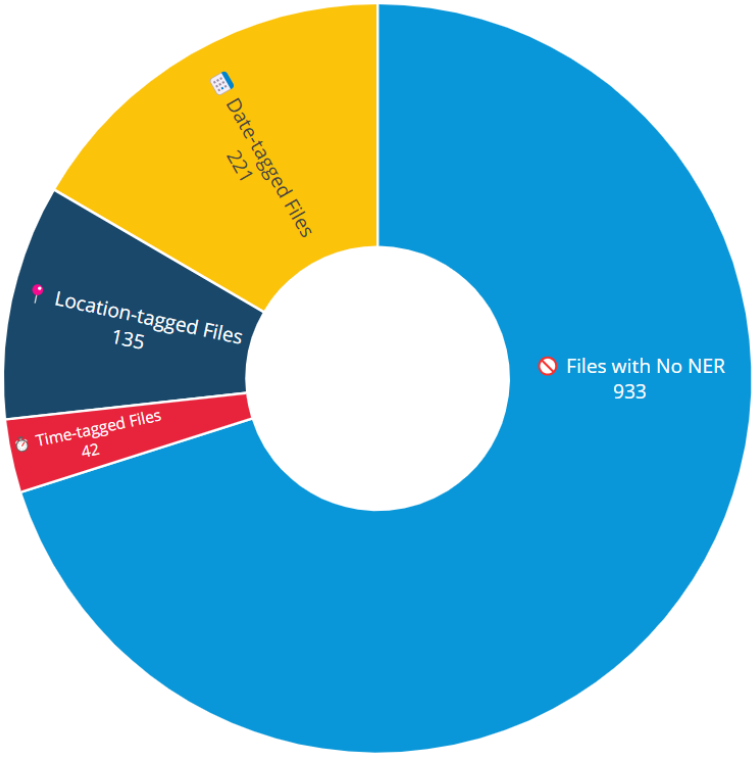

Figure 16: Number NER Representations in the files

This chapter focuses on annotating EU legislative files using Symbolic AI. Each annotation includes a "definition heading" and a "definition body." The dataset comprises 15,082 files of EU legislation, which have been pre-processed into AKN file format. For annotation, the study is divided into two scenarios. The first scenario annotates definitions within articles under specific "heading" tags. In the second scenario, independent *Delimiting Definition*s are annotated wherever they appear in the files. The annotation algorithms are designed and implemented in Python, using the ElementTree library to handle XML files. The annotation process begins by using rule-based mining to detect targeted text. Once

a targeted text is identified, the heading and body of each definition are annotated with tags such as “def” and “defBody.” The annotated information is then listed, and the AKN files are validated through indentation. Applying the first algorithm, 899 files were successfully annotated with relevant content, while in the second scenario, 1,272 files were annotated. The annotation of definitions is verified by comparing it with the best manually annotated files, and the calculated Kappa value is 1, confirming that the performed annotation is perfect. Finally, 11705 definitions were annotated, with 92.88% identified within articles under the tag heading containing the inner text "Definition/definitions".

Chapter 4
Sustainable Development Goals Classification

4 Introduction

This chapter discusses the classification of EU legislative documents into Sustainable Development Goals (SDGs) using Artificial Intelligence (AI). As outlined in Chapter 1, the SDGs consist of 17 predefined Goals, each accompanied by specific Targets that contribute to achieving the respective Goal. There are 169 Targets in total, and each Target is further associated with specific Indicators. The structure of SDG Goals, Targets, and Indicators is illustrated in Figure 17. This chapter discusses the classification of SDGs at the Goals and Targets level, as well as the results of the classification.

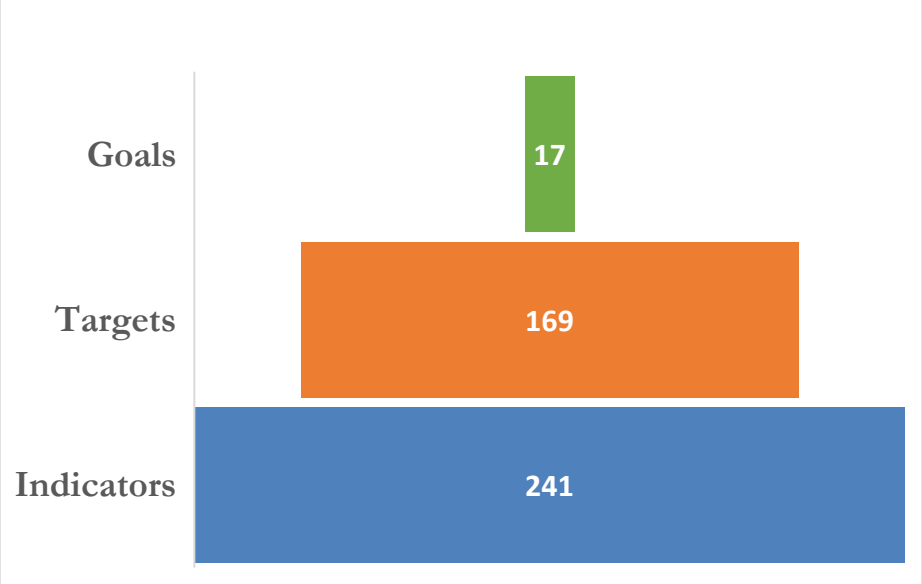

Figure 17: Sustainable Development Goals

4.1 State of the Art

Legal text classification involves identifying and categorizing legal entities based on their associations. It aims to determine a legal text's category based on its context and relevance or association [119]. This process plays a significant role in developing intelligent or AI-based

systems, which are particularly beneficial for judiciary personnel. With the exponential increase in legal and judicial documents, finding previous similar cases for argumentation has become challenging. AI-based systems address this issue by classifying legal documents and cases according to their associations. This classification of legal acts, cases, or documents facilitates the extraction of specific types of legal documents. Such classifications of legal texts have traditionally relied on Machine Learning or Natural Language Processing techniques. This section presents some past studies on legal text and document classification.

Legal text and legal document classification have been investigated with different Machine Learning, Deep Learning, and Natural Language Processing strategies [120]. The [121] identified argumentative structure, function, and propositions from the European Court of Human Rights (ECHR) and [119] classified an Italian legal text into a relevant domain. The [122] predicted the case where an ECHR legal document was from, what area of law the case was about, and the date the ruling was issued, [123] also focused on the prediction of legal judgments from ECHR cases. Machine Learning was used to predict the ruling of the French Supreme Court and to predict the belonging of cases within the law area [124] and [125] employed models based on RNN to identify necessary parts and realize them in Japanese legal texts. Evidence extraction from Chinese court documents was done using an information extraction task that integrated with the classification of legal articles [126]. Legal text classification also includes the specific tasks in the classification of the law area [119], argumentation mining [121], prediction of court decisions [120] and the identification of rulings of courts [122]. Further studies related to text and legal document classification are presented below in detail.

In [127], legal documents were classified using Neural Networks. The dataset used in this study was based on SCOTUS legal opinion consisting of 8419 US Supreme Court opinions from 1946-2016. The corpus belonging to 15 legal categories was split into 80%, 10% and 10% for the model's training, testing and validation, respectively. Noisy and special characters were removed in the initial preprocessing of the dataset.

Latent Dirichlet Allocation (LDA) was used to organize the legal documents based on probability extraction. After organizing legal documents into 15 legal issues and 279 legal subtopics. For the model design, the Logistic Regression was applied using Doc2Vec, Support Vector Machine was applied with a Bag of Words (BOW) and a Convolutional Neural Network with Word2vec. In terms of classification, Word2vec outperforms the other classifiers in the classification of 15 legal issues and 279 legal subcategories, with 72.4% and 31.9% accuracy, respectively. In comparison, CNN+Word2vec performs 65.9% and 14.7% accuracy, respectively.

In [128] automated legal text classification was presented using a famous ML algorithm named Random Forest and DL. The dataset that was used in this study was based on US legal documents from 50 different categories. The dataset was consistent with 30.0000 US case legal documents. For the feature extraction from legal documents, the TF/IDF technique was implemented, and for the feature selection, the Principal Component Analysis (PCA) technique was implemented. For DL, the pretrained word embedding techniques were used for SigmaLaw, a subset of the dataset. For the legal text classification, firstly, the terms were extracted from the legal documents using POS tagging with NLTK, where POS tagger achieved 94% accuracy. After the POS tagging, the term filtration was performed using a filtering approach, which calculates the relevance of the term using domain consensus and pertinence. After-term weight calculations have a distinct role in the legal domain for representing specific information. RF was implemented for classification purposes from ML, and BiLSM was used from DL TextCNN. The best result was 84.49%, which was presented in this study using Domain concept + Random Forest. This study also concluded that the domain concept-based random forests classifier increases the results up to 35.26% in terms of F-score.

In [129] a framework was proposed to classify the explainable text in legal documents. For this purpose, the dataset consisted of documents

collected from different sources, i.e., Microsoft Office documents, PDFs, email documents, and other text-based documents. As a result, 688294 documents were collected that were classified by the attorneys into two classes: i: responsive and ii: not responsive. Of them, 41739 documents belonged to the responsive class, and the rest, 646555, belonged to the non-responsive class. The dataset mean length of the document was 52 words, and the standard deviation was 112.5 words. For the model design that was designed using Logistic Regression, the dataset was split into 3 different snippet sizes: 50, 100 and 200 snippets. For the responsive document, the best results were achieved in 200 snippets, with 84.17% in recall. This study handles the unbalanced classification of text in legal document reviews.

In [130] the work done was the same as in the previous one. The authors classified the legal documents into responsive documents and non-responsive documents. For this purpose, this study collects the dataset from Legal Track at the Text Retrieval Conference (TREC). The dataset consists of three datasets called D1, D2, and D3. D1 consists of 2500 general-purpose trading documents, of which 1040 documents belong to the responsive documents and the remaining 1460 belong to the not responsive class documents. D2 has concerns about the legality and illegality of trading of financial products with 2102 total number of legal documents. Of these, 238 belonged to the responsive class, whereas the rest of the 1864 documents belonged to the documents that were not responsive class. D3 has a total of 2199 documents related to the environmental effects of the company activities. 245 documents belonged to the responsive class document, and the remaining 1954 documents belonged to the responsive document class. CNN, RNN, LSTM and GRU were implemented for the model design. At 75% of Recall, CNN outperformed the other classifiers with 92.11% Precision on D1, 85.99% on D2 and 50.27% on D3, respectively. At 90% of Recall, CNN outperformed the other classifiers with 84.59% Precision at D1, 67.08% on D2 and 34.85% Precision on D3.

In [131] the comparison of CNN on the Text classification of legal documents with the other ML algorithms, i.e., Support Vector Machine,

Random Forest, and Logistic Regression. The dataset splits into 4 sets; in set A, out of 410954 documents, 81234 documents belonged to responsive class documents, and the remaining 329630 were not responsive documents. 19.8% of documents were responsive from set A. In set B, 1570956 documents, 408897 were responsive documents, 1162059 were not responsive, and the percentage of responsive documents was 26.0. In set C, there were 492318 documents, of which 201147 were responsive, and 291171 were not responsive documents, with 40.1% of responsive documents. Set D has 308738 documents with 15.1% of responsive documents, of which 46644 were responsive documents, and the rest, 262094 were not responsive. The datasets were preprocessed by applying different preprocessing techniques, i.e., token filtering, tokenization, stemming, N-gram generation and feature selection. At 75% Precision rate, CNN outperforms the other classifiers.

In [132] a dataset based on the multi-lingual was proposed for the classification of legal documents into multi-label. The proposed dataset comprises 65000 EU law documents that belong to 23 EU languages i.e., English, German, French, Italian, Spanish, and Polish. The proposed dataset was divided into several levels based on labels, i.e., Level 1 has 21 labels, Level 2 has 127 labels, and the total number of labels in the dataset was 7390. Several pre-trained models were used to classify legal documents into multi-label documents, i.e., BERT and XML-ROBERTA. In [133] classification of large and unstructured legal documents using hierarchical and Large Language Models (LLM). The dataset used in this study was ILDC (Indian Legal Documents Corpus) which contained 39898 unstructured legal documents in the English Language. These documents were the case transcripts of the Supreme Court of India. The second dataset was LexGLUE (Legal General Language Understanding Evaluation), a benchmark legal document dataset. Multi-stage Encoder-based Supervised with-clustering (MESc) and LLMs (i.e., BERT, GPT-J, and GPT-Neo) were used for the classification purposes.

The BERT outperformed the others with an accuracy and F-score of 84.15%.

In [134] Multilabel Text Classification Convolutional Neural Network (MLTCNN) was proposed for the classification of legal documents in the Chinese language. For the dataset, 143 legal articles were scanned, and text was extracted by applying OCR (Optical Character Recognition) with a sensor device named “CCD” (Charge-Coupled Device). Different preprocessing techniques were adopted for the preprocessing of the data. Word segmentation was performed to segment the words according to the Chinese character sequence. After that, the stop words were removed and after the vocabulary generation, the proposed MLTCNN model was applied that was based on the TextCNN. MLTCNN outperformed the other classifiers with 98.07% Precision, 99.61% Recall and 98.84% F/score.

In [135] classification of legal documents was done using classical Machine Learning and Deep Learning algorithms. The legal documents dataset was used in this study. Word embeddings were performed by using Word2Vec and Glove transfer learning. The dataset was large enough, so the dataset was split into 4 chunks. On chunk D of the dataset, the model outperformed with the highest number of accuracies, where SVM performed 92.53% in terms of accuracy on train set 2. Where the CNN performance on the same dataset was 91.58%. The authors highlighted the challenges in the classification of legal documents, the text's length, and the text's sequence. If it is difficult to keep the relevant part of the training, don’t chop off the relevant text before feeding for training of the model.

In [120], legal text classification was performed by applying a famous ML algorithm, i.e., SVM. The dataset for legal text classification was collected from the Court of Cassation and the French Supreme Court. The dataset consists of 131830 legal documents. These documents were in XML file format. As the result of the preprocessing of data all duplicates and incomplete entries, 4965 documents were removed. Now, the final dataset has 126865 documents. This paper tries to classify legal documents in three ways: 1: predicting rulings based on the case

description, 2: predicting the law area of rulings and cases, and 3: predicting the time of issuance of rulings based on the case description. The prediction of the ruling model outperforms the classification results with 98.6% both accuracy and F-score. Regarding the prediction of the law area, the model correctly predicted 96.8% of documents, and the value of the F-score was 95.5%. On the prediction of issuance of document, the model performs 87% with accuracy and F-score.

In [136] legal document classification was done using Natural Language Processing. The dataset that was in this study consists of 17740 documents that were collected from 18 different fields of the law, including petitions from the year 2016-2019. The dataset was further preprocessed before classification by applying lowercasing, tokenization, punctuation removal, POS tagging, and normalization based on names and numerals. Further features were extracted using TF/IDF. Linear, boosted trees and neural network-based algorithms were implemented for the classification. The linear family used Logistic Regression and Support Vector Machines for legal document classification. From the boosted trees, random forest, gradient boost trees, and neural family, text CNN and RNN were implemented. The dataset was split into 90:10 for the model's training and testing. RNN using Long-Short Term Memory outperforms the other classifiers by accurately classifying 90% of documents. In comparison, the F-score was 85%.

In [137] the classification of lengthy legal documents was done. For this purpose, the dataset was collected by downloading the legal documents from the U.S. Securities and Exchange Commission (SEC) public database, which consists of 28445 legal documents. For the classification, Bidirectional LSTM and SVM were implemented by splitting the data into 70% for the training of the model. In contrast, the remaining 30% of the data was used to test the model. In evaluation, the LSTM performs 97.85% F-score on the testing of the model and 98.11% on the validation of the model. Where SVM outperforms the LSTM with 97.97% testing and 98.20% validation F-score, these scores were

calculated on the 3rd chunk of the dataset that was split due to the machine's limited specification, i.e., memory.

In [138] text from legal documents were classified. The dataset for the text classification in legal documents was collected from the State of Ceará, in Brazil, and consists of lawsuits from the Court of Justice. Of the 16668 petitions, 7103 lawsuits were selected for the classification task that was defined by the National Council of Justice of Brazil. Optical Character Recognition (OCR) was used to extract text from legal documents. Random forest, support vector machine, extreme gradient boosting, and bidirectional encoder representations from transformers are used for the legal text classification. In terms of classification, the BERT outperforms the other classifiers in terms of embeddings of the lawsuit text with a citation with an 88% F-score.

In [139] a novel dataset, which consists of Brazil's Supreme Court legal documents, was proposed, named VICTOR. The proposed dataset consists of 45000 appeals, including 692000 documents that have 4.6 million pages. The study was further extended with the classification of legal documents. For the classification of legal documents, the model using SVM, NB, BiLSTM and CNN NV was trained on 70% of the total dataset; the remaining 15% was used to test and 15% to validate the model. The dataset was split into three different versions, i.e., Big-VICTOR (BVic), Medium-VICTOR (MVic) and Small-VICTOR (SVic). On SVic, CNN outperforms the other algorithms with 86.43%, and XGBoost outperforms the other algorithms with 88.87% F-score on the classification of legal documents using the SVic dataset.

In [140] an open-source online tool (OSDGs tool) was proposed by the UNDP SDGs AI (Artificial Intelligence) lab to classify the text data into the SDGS using ontology engineering. OSDGs tool allows users to tag SDGs documents using text, summary, abstract, or the DOI number. In [141] a multilingual tool was proposed to classify text data according to the United Nations Sustainable Development Goals. 15 different languages supported that tool. The languages that were supported by the OSDGS2.0 tool are Arabic, English, Dutch, Danish, French, Finnish, Italian, Polish, German, Korean, Spanish, Portuguese, Turkish and

Russian. They developed a web-based open-source tool. After pasting the text in 15 different languages in the search bar, it can be classified into the available SDGs in the text. It also can classify the documents into the SDGs. For this purpose, they developed an OSDGs community dataset annotated by the UN volunteers (who all graduated and were aware of the task of SDGs classification, all the volunteers graduated). After training the 900 online UN volunteers recruited through the Unified Volunteering Platform from distinct parts of the World, they made a web-based platform for annotation with volunteers. Where the volunteers just read the text and annotate it into the 17 SDGs. This task was done by the volunteers manually, and the volunteers had to read the text and annotate it into the SDGs or SDGs. After the annotation, the OSDGs community dataset was created. Later, they used the OSDGs dataset to train the Machine Learning models. After the training, the model predicts the SDGs found in the text or the document. After the SDGs prediction, the output is verified or validated by using ontology or keyword mapping techniques. The OSDGS2.0 tool was aimed to navigate the SDG-related ambiguities present in the document or the text by simply providing it to the application that identifies SDG-relevant content in the document or the text.

In [142] an SDG-Meter was proposed for the automatic text classification of the Sustainable Development Goals using the BERT model. SDG-Meter also allows an online web-based tool for the classification of text into the SDGs. The tool can classify the given text into all possible SDGs (one or more) present in the text. It allows users to put text up to 512 words into SDG-Meter, and after applying the tool, they can see the results of classification into related SDGs. The dataset used in this study consists of 6000 texts that were collected from various semantic structures (research articles, official reports, news, etc.). The dataset was labelled using the BERT model to annotate the text into the 17 SDGs. The average length of the text is 374 words. The final dataset has 17 labels (having all SDGs). The collected dataset has an unbalanced class

problem. 80% of the set was split into an 80:20. 80% was used for the training of the model, and the remaining 20% was used for testing the model. At the testing dataset, the proposed model performed with 94% accuracy in the classification of text into the SDGs. The dropout method was used to overcome the issue of overfitting. The limitation of the SDG-Meter is that it tries to link the given text with SDGs if the SDGs are not even present in the text. The authors claimed that the SDG-Meter faces that problem due to the small amount of training dataset. The authors are confident in introducing a new model with the name of "SMITH" with more training data and better results. The length of the text also increases by a maximum of up to 2048 words in the upcoming model "SMITH".

In [143] comparison of results of multi-label text classification of research-based scientific articles that were aligned with the SDGs between the DistliBERT and SVM. For the study, they collected a dataset from bibliographic resources. From each scientific article, three features were extracted for this study, i.e., the title of the research article, the abstract, and the labels from 17 SDGs presented in the study. The dataset was created by collecting research articles that were published in 2018. The selected domain was Dominion Organic Agriculture 3.0. The dataset consists of 31434 instances. After collecting the dataset, it was preprocessed for further use by using NLTK and scikit-learn libraries for classification. In preprocessing, stop words were removed from the dataset, and tokenization and stemming were also performed. The rest of the features were extracted using the Term Frequency-inverse Document Frequency (TF-IDF). Later, the dataset was categorized into 5 different scenarios. In 1st scenario, there was imbalanced data with all 17 labels; in 2nd scenario, there was imbalanced data with 11 SDGS that were greater than 1000 instances. In the 3rd scenario, the balanced dataset had an equal number of instances in 11 SDGS; in the 4th scenario, the extremely imbalanced dataset was from one label to another 10 labels. In the 5th scenario, instances with only one SDG label have multiclass labels. After the preprocessing, they developed two types of models for the classification of documents into the SDGs; one was based on the ML-based classification algorithm, i.e., Multi-class SVM, and the second was based

on the BERT. For the model design, the dataset was split into 2:1 for the training and testing of the model, respectively. After the classification, the models were evaluated using Accuracy, F1-score and Hamming loss. The SVM outperformed the DistilBERT in all five scenarios in terms of accuracy. The maximum accuracy was achieved using the dataset of the 5th scenario, where the SVM performed 0.893%, and DistilBERT performed 0.875% in terms of accuracy. The authors proposed that further study be extended by defining and adjusting hyperparameters of both models by quantifying and improving their performance.

In [144] classification model was proposed for the classification of the Creditor Reporting System (CFR) dataset into the SDG. CFR is a key source of datasets for the monitoring and evaluation of aid flows and is in line with the 17 Goals of SDG. The CFR was further preprocessed with the cleaning of text descriptions, and then the data was split into a 3:1 ratio for training and testing of the model, respectively. 45405 documents were used for the training of the model, and the remaining 15135 documents were used to predict and check the model's performance using different machine learning algorithms. This study used Random Forest, TextCNN, BiLSTM, BERT and ELECTRA to predict documents into 17 SDGs. Upon evaluation of the model, the ELECTRA outperformed, with an accuracy of 89.71%. The model was also evaluated with the F1-score. This study was further implemented on a sector level, i.e., Education, Government & Civil Society, Communications, Agriculture, Forestry, Fishing and Health. At the sector level, the model achieved maximum accuracy in Communication with a score of 98.07%, and the value of the F1 score was 98.25%.

In [145] multi-label text classification models were compared based on published research articles. The articles were classified into the 17 Sustainable Development Goals of the United Nations. The dataset was gathered from different sources, i.e., Scopus, Microsoft Academia and Web of Science, that contained 180852 scientific research articles from January 2015 to August 2021 with the title and abstract from the

agriculture domain. The dataset was gathered in a CSV file. For dataset creation, a multi-label binarizer was used that converted the dataset into the binary matrix data frame of 17 SDGs class labels. For the preprocessing of the dataset, stop words were removed, text was changed into lowercase, the symbols and contractions were removed, and features were extracted using the TF/IDF and tokenization method. This study used Naïve Bayes, Logistic Regression, Support Vector Machine, and Random Forest for the multi-label classification of scientific research articles into 17 SDGs Goals. For the model, the dataset was split into a 2:1 ratio for training and testing of the model, respectively, and the rest of the results were computed using evaluation measures, e.g., accuracy, F1-score, and hamming loss. This study achieved the best results on the 2018 dataset, at which SVM with LP performed with the highest accuracy, which was 83%. They proposed that SVM is the best ML algorithm for the classification of multi-class problems.

In [146] the United Nations Department of Economics and Social Affairs proposed a Knowledge Organization System for the United Nations Sustainable Development Goals. It provides an online tool that links text to the SDGs using an ontology-based technique. In [147] several studies were proposed to classify the SDGs by the Organization for Economic Co-operation and Development (OECD). A method was proposed to link financial aid input with the development of SDGs. Another methodology was proposed to find interlinkages of multiple SDGs. A tool named SDGs tracker was developed using ULMFiT (Universal Language Model Fine-Tunning for Text Classification) and XGBoost (Boosted Gradient) to link the project description of financial contributions to the addressed SDGs. After the classification of SDGs is performed, the output is verified by manual experts. A literature review of the classification is shown in Table 7.

Table 7: Literature Review on the Classification of SDG

AUTHOR	DOMAIN	APPROACH	DATASET	DATA SIZE	F/SCORE	ACCURACY	TECHNIQUE
Guisiano	Text Classification	SDG-Meter	Research Articles, Official Reports, News, etc	6000 instances	N-A	94%	BERT
Morales-Hernández	Multi-Label Text Classification	ML and NLP	World-Class Lattices (WCL)	31434 instances	N-A	1: 87.5 2: 89.3	1: DistliBERT & 2: SVM
Pukelis	Text Classification	ML	OSDG	37575	N-A	N-A	OSDGS2.0 tool
Pukelis	Articles & Text Classification	NLP	5 Different	14000 Key Terms	N-A	N-A	Ontology Engineering
Pincet	SDGS Linking	SDG STracker	Text Data	N-A	N-A	N-A	ULMFiT, XGBoost
Morales-Hernández	Multi-Label Text	ML	Research Articles	180852 Scientific Articles	90.1	86.8	NB SVM LR RF
Joshi	SDGS Linking	NLP	UN Documents	5000 Synsets	NA	NA	Ontology-based Technique
Lee	Article Classification	ML	Creditor Reporting System (CFR)	60540 documents	87.40 ELECTRA	89.71 ELECTRA	RF, TextCNN, BiLSTM, BERT & ELECTRA

4.2 Methodology

This section discusses the methodology for classifying Sustainable Development Goals at the Goals and Targets level. The methodology for developing a classification model is divided into four phases. The first phase discusses dataset acquisition and its preprocessing in detail. The second phase explores model design and experimentation by applying Machine Learning algorithms. The results are analyzed in detail in the third phase to identify the best classification model for linking the EU legislation with SDGs at the Goal and Target levels. The rest of these phases of the methodology are discussed in detail in the coming sections.

4.2.1 Data Acquisition

The classification of EU legislative documents into the SDGs aims to identify the actions found in them and establish their links to the corresponding SDGs. To classify the EU legislative document into the SDGs, the data acquisition is performed in two steps: the first is about the LEOS (Legislation Editing Open Software) dataset [39], which is used in the annotation of the *Definition*s process (see Chapter 3), and the second dataset, which is scraped from KnowSDGs (Knowledge Base for the Sustainable Development Goals). The KnowSDGs dataset consists of previous initiatives from 2015-2019 [148]. The rest of the details are discussed in the sections below:

4.2.1.1 LEOS

The LEOS dataset was based on EUR-Lex documents that had already been converted into AKN format. By applying AI-based algorithms, *Definition*s were detected and annotated in these legislative documents. The resultant 1272 legislative files were found to contain annotated material, and in these 1272 legislative files, 11,705 *Definition*s were successfully annotated (the details are discussed in the annotation Chapter 3). These 1,272 legislative documents were further used in the classification process to identify and link the present actions in these EU legislative documents at the Goals and Targets level of the SDGs. Since these 1,272

legislative documents were not labelled with SDGs, they were only processed to identify the SDGs and to link them with the SDGs at the Targets and Goals level. An annotated dataset is scraped from KnowSDGs to develop a classification model. The details are given in the section below.

4.2.1.2 Annotated Dataset

The annotated dataset is scraped from the KnowSDGs platform that provides policy mapping tools to demonstrate how EU policies address the SDGs. The platform offers two different categories of documents: (i) Preparatory documents and (ii) Legal acts. These documents belong to two different periods: previous initiatives and current initiatives. The previous initiatives cover the years 2015–2019 and are annotated manually, while the current initiatives cover 2019–2024 and are annotated using an AI tool [148] [149].

For this study, the previous initiatives were chosen because these documents are manually annotated, which increases the likelihood of correctness [130]. Therefore, the previous initiatives were selected and scraped using a Python Script to develop a classification model for classifying the SDGs.

4.2.1.2.1 Pre-Processing

The scraped dataset consists of previous initiatives spanning six different legislative types: Directives, Decisions, Regulations, Recommendations, Declarations, and Resolutions. The Regulations are represented with 'R' in the file names, Directives with 'L,' and Decisions with 'D.' A total of 8,803 records were scraped from the KnowSDGs platform, encompassing six types of documents in EU legislation.

This study, however, focuses only on Directives ('L') and Regulations ('R'). These specific records were separated from the rest. This results in a dataset of 2,790 records in an XLSX file being separated. Related to

the 'L' and 'R' types of legislation. After separating the targeted files in the XLSX file with their CELEX numbers, which are already labelled with SDG Goals and Targets. The text from the files with the same CELEX numbers was extracted from the LEOS files and added to the XLSX file. In the first column, the file contains the CELEX, which is the identification number of each file, as discussed in Section 2.3.

The second column includes the text of each file's articles. The third column includes the text of the first four articles. The fourth column includes the preambles of each corresponding file. The fifth column presents each file's SDGs labels (Goals and Targets level). Figure 18 shows a visual representation of the data in XLSX.

	A	B	C	D	E
1	CELEX	ARTICLE	ARTICLE4	PREAMBLE	CLASS
2	32016R1644	Article 1 The name "Φάβα Φε	Article 1 The name "Φάβα Φενε	THE EUROPEAN COMMISSION, Havi	2.3
3	32016R1646	Article 1 Main indices The ma	Article 1 Main indices The main i	THE EUROPEAN COMMISSION, Havi	2.3, 17.1
4	32016R1649	Article 1 This Regulation estat	Article 1 This Regulation establis	THE EUROPEAN COMMISSION, Havi	8.2, 9.1, 9.4, 9.5, 11.2
5	32016R1661	Article 1 Annex I to Regulatior	Article 1 Annex I to Regulation (E	THE COUNCIL OF THE EUROPEAN UI	8.1, 16.3, 16.4, 16.1
6	32016R1662	Article 1 The name "Sicilia" (P	Article 1 The name "Sicilia" (PGI)	THE EUROPEAN COMMISSION, Havi	2.3
7	32016R1675	Article 1 The list of third-coun	Article 1 The list of third-country	THE EUROPEAN COMMISSION, Havi	16.4
8	32016R1676	Article 1 The name "Saucissor	Article 1 The name "Saucisson se	THE EUROPEAN COMMISSION, Havi	2.3
9	32016R1683	Article 1 Annex I to Regulatior	Article 1 Annex I to Regulation (E	THE EUROPEAN COMMISSION, Havi	16.9
10	32016R1686	Article 1 For the purpose of th	Article 1 For the purpose of this I	THE COUNCIL OF THE EUROPEAN UI	1.2, 1.4, 5.3, 16.1, 16.2

Figure 18: Visual Representation of Labelled Data

Out of the 2,790 records consisting of 'L' and 'R,' six files (32015R0864, 32015R0523, 32015R0341, 32015R0228, 32015R0220, 32015R0207) did not contain any text in preambles or in the articles. These six records were removed from the XLSX file. The remaining (2784) dataset was further refined using various data preprocessing techniques in Data Mining.

This study incorporated three preprocessing techniques to improve classification results: lowercasing, Removing Stop Words, and Removing Short Words. These three techniques were selected after experimenting with various preprocessing methods, including Lowercasing, Removing Stop Words, Removing Short Words, Removing Special Characters and punctuations, Stemming, and Lemmatization. Different combinations of these techniques were tested, and based on the results, the three techniques mentioned earlier were chosen for their effectiveness.

4.2.1.2.2 Splitting Dataset

After preprocessing the dataset, 2,784 records remained, each containing classified text of legislative files with SDGs. This labelled dataset has an imbalanced class problem. The imbalanced distribution of classes in the training or testing set badly affects the classification results [150]. To overcome this problem, the data is split into 2:1 using a stratified sampling method in the training and testing datasets. Stratified sampling is a method that aims to preserve the proportions of different categories or classes from the original data in both the training and testing datasets [151].

After splitting the dataset, the training dataset consisted of 1,856 records, while the testing dataset contained 928 records. These datasets were used to train and evaluate the classification model.

4.2.1.2.3 Oversampling of Training Dataset

As discussed in the above section, the scraped dataset of previous initiatives exhibits a class imbalance issue. Some classes appear only once or twice. While this did not cause problems during training, it led to warnings for unknown classes when these infrequent classes appeared in the test set. To address this issue, the training dataset was oversampled after the dataset was split into training and testing sets using a stratified sampling method. During oversampling, the minimal classes were scaled up to have a minimum presence of 20 records per class (minimum presence of any SDGs Target is scaled up to 20 in the training dataset). After oversampling, the total number of records in the training set increased to 2,497 for the Target-level classification. The original dataset is used after being stratified into training and testing datasets for the classification of SDGs.

4.3 Modelling and Experimentation Design

This section provides a detailed discussion of the model design and experimentation aimed at developing a model for SDGs classification. Mainly, the classification is divided into three categories:

i. Binary Classification
ii. Polynomial Classification
iii. Multilabel Classification

In binary classification, the data is categorized into one of two classes. In polynomial classification, an entity is classified into one of multiple possible classes. In a multilabel classification problem, entities can belong to multiple classes simultaneously [152].

This study specifically addresses the multilabel classification problem. The model was initially designed to achieve multilabel classification of SDGs at the Goals and Targets level.

4.3.1 Model Design

This study utilizes an annotated dataset based on previous initiatives scraped from the KnowSDGs online platform to design the classification model. After preprocessing, the KnowSDGs dataset is split 2:1 using the stratified sampling method. Two-thirds of the dataset is allocated for model training, and one-third is reserved for testing and evaluating the ML model.

The model, based on the ML algorithms, is trained on 67% of the total data and subsequently tested and evaluated on the remaining 33% of the dataset. The model's performance is assessed using both the weighted and macro F-scores. The weighted F-score measures the overall performance of the model and is calculated as the average F-score across all classes.

The macro F-score is used due to the class distribution of the dataset. Since the dataset is unbalanced, with some classes having very few samples, the macro F-score provides a more representative evaluation. Unlike the weighted F-score, the macro F-score accounts for class imbalance by treating each class equally when calculating the final score. The

model is evaluated using both Weighted and macro F-score, while other evaluation metrics are also presented, i.e., Accuracy, weighted Recall, weighted Precision, macro Recall and macro Precision.

After developing and testing the model, the trained model will be saved and utilized to link the LEOS dataset (legislative files of the EU) with SDGs. After saving the model, the EU legislative documents are given to the model to link the SDGs at both the Goals and Target levels (see Chapter 5 for the details). Figure 19 illustrates the model design used in this study.

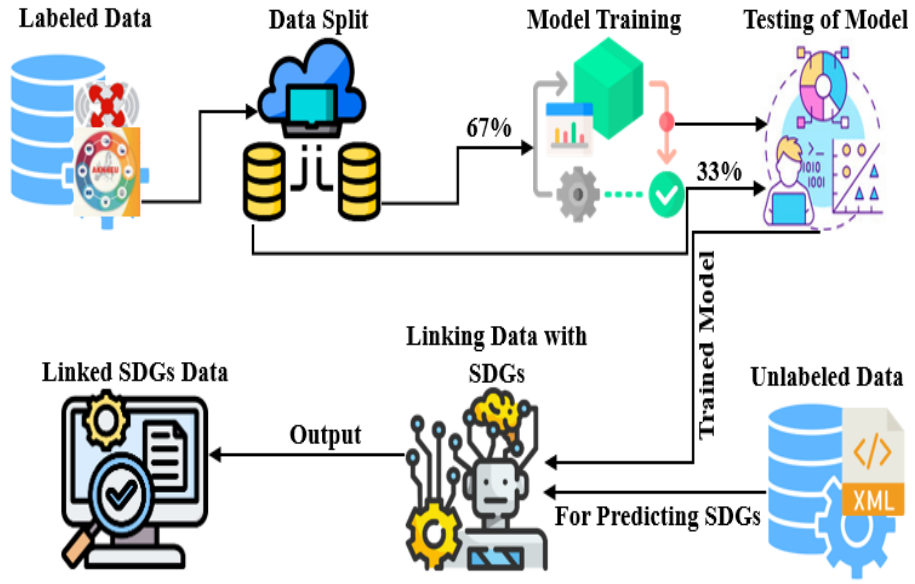

Figure 19: Model Design

4.3.2 Experimentation Design

For the experimentation design, various Machine Learning and Deep Learning algorithms are implemented to identify the most effective approach for SDGs classification. The classifiers evaluated in this study are:

i. Naïve Bayes
ii. Random Forest
iii. Text Convolutional Neural Network (TextCNN)
iv. Recurrent Neural Network (RNN)
v. BiLSTM
vi. Logistic Regression
vii. Support Vector Machine
viii. K-Nearest Neighbours

These algorithms are tested based on good results, the SVM and KNN are chosen for the final implementation. The representation of results of different algorithms of ML that were tried on the classification of Sustainable Development Goals by taking the text of the first four articles and the preamble text is shown in Table 8.

Table 8: Representation of Results of Different Algorithms of ML Tried on the Classification of SDGs Goals by Taking the Text of First Four Articles and the Preambles

	ACC	**Precision**	**Recall**	**F-score**
NB	30.00	98.99	13.51	23.78
Rf	37.96	82.19	27.68	41.41
TextCNN	38.28	58.99	33.18	38.43
RNN	00.00	01.26	00.09	00.17
BiLSTM	00.22	16.60	01.11	01.19
LR	49.81	48.56	55.81	49.62
SVM	**56.25**	**77.48**	**62.26**	**67.86**
KNN	**55.39**	**75.14**	**59.93**	**64.99**

4.3.2.1 Support Vector Machine

Support Vector Machine (SVM) is a supervised learning algorithm that analyzes documents and recognizes patterns for classification. It can

efficiently solve multilabel problems and categorize multi-dimensional documents or data using One-Versus-Rest (OVR). SVM represents data as points in space and creates well-defined margins between the classes to ensure robust classification decisions. The polynomial kernel is implemented to handle multilabel classes efficiently [152] [153].

Advantages:

- SVM is a powerful classifier that handles high-dimensional data effectively.
- It performs well on semi-structured and unstructured data.
- SVM offers advantages in terms of memory efficiency and computational speed.

Disadvantages:

- SVM requires a long training time for large datasets.
- SVM is computationally expensive for large datasets.
- It struggles with data that contains outliers.

4.3.2.2 K-Nearest Neighbors

The K-Nearest Neighbor classifier was first introduced in the mid-1950s and did not gain popularity until the 1960s. It is based on instance-based learning used for the classification and regression task. The KNN algorithm assumes all the instances corresponding to the points in the n-dimensional space. It can simply compare the test instances given with similar training instances defined in terms of the standard Manhattan distance, Hamming distance, Minkowski distance, and Euclidean distance. When KNN is applied to the multilabel classification problem, it is adopted to predict multilabel for each entity, and each entity is associated with the multilabel [154] [155].

Advantages:

- KNN easily handles the complex relationships.
- KNN performs well on pattern recognition problems.

- KNN is simple to understand and easy to implement.

Disadvantages:

- KNN is computationally expensive for large datasets.
- KNN does not deal well with missing values.
- KNN is a lazy learner because it memorizes the training data.

After the final selection of ML algorithms (SVM and KNN) for multi-label classification problems, feature extraction, dimensionality reduction, and rare event prediction techniques were implemented to improve the model's performance. The remaining details are provided in the following sections.

4.3.2.3 Feature extraction

Feature extraction involves converting text data into a structured format suitable for Machine Learning models. This study experimented with TF/IDF (Term Frequency-Inverse Document Frequency), Word2Vec embeddings, and N-grams (Unigrams, Bigrams, and Trigrams) with TF/IDF. Results indicate that TF/IDF alone performs better than the combinations with other techniques. Therefore, TF/IDF is chosen for final implementation [156].

4.3.2.4 Dimensionality Reduction

Dimensionality Reduction reduces the number of features while retaining critical information. This study tested Principal Component Analysis (PCA), t-SNE (t-distributed Stochastic Neighbor Embedding), and UMAP (Uniform Manifold Approximation and Projection) for the dimensionality reduction of the text data. Among these, PCA provided superior results and was selected for the final implementation [157] [158].

4.3.2.5 Rare Event Prediction

Rare event prediction focuses on identifying and forecasting entities with low frequency but high impact. To address class imbalance,

oversampling is applied to the training (separated from the test set) dataset (see section 4.2.1.2.3). This process ensures a minimum presence of instances 20 times for any class in the training data.

Class Weights are also incorporated to handle rare event prediction and improve results, particularly for imbalanced data. SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling) were also tried. Still, these techniques are limited to binomial and polynomial classification problems, and this study deals with multilabel classes. KNN does not support Class Weights, so Class Weights are applied with SVM only, resulting in improved performance.

After evaluating various techniques, TF/IDF, PCA, and Class Weights were selected based on their enhanced results to incorporate with the final implementation. All the above-mentioned techniques are integrated with SVM. For KNN, only TF/IDF and PCA are implemented; KNN does not support Class Weights.

Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) are applied to the multilabel classification of SDGs. The experiment is designed using four different scenarios based on the 17 Goals and 169 Targets of the SDGs. The details of the designed experiment are given below:

i. 1st, with the first 4 articles
ii. 2nd, with all the articles
iii. 3rd, with the first 4 articles and the preambles
iv. 4th, with all the articles and all the preambles

In the first experiment, the model is trained using the text of the first four articles of each file as input variables and tested on the same set of first four articles. In the second experiment, the model is trained by incorporating the text of all the articles and tested on the entire set of articles. In the third scenario, the model is trained and tested using the first four articles' and preambles texts. Finally, in the fourth scenario, the model is trained on all articles' text along with preambles text and tested

on the complete dataset, including all articles and their corresponding preambles.

4.4 Results

As discussed in the previous section, the best-performing preprocessing techniques were chosen based on their effectiveness for classification. At both the Goal and Target levels, the SDGs classification results were conducted and evaluated using the four experimental scenarios described earlier.

The classification was conducted in two stages. First, it was performed on the 17 Goals of the SDGs using the four abovementioned scenarios. Subsequently, it was extended to the 169 Targets. The detailed results of the SDGs classification for both the Goals and Targets levels are presented in the following coming sections.

4.4.1 Classification of SDG Goals

This section presents the results of classifying SDGs at the Goals level. The classification was performed using ML algorithms: Support Vector Machine (SVM) and K-nearest Neighbors (KNN) for multilabel problems.

4.4.1.1 First 4 articles

For the initial experiment, the text data from the first article is used as input variables to develop a classification model. The model is developed using four consecutive scenarios: using SVM, SVM by incorporating PCA, SVM with Class Weights, and SVM with both PCA and Class Weights. After model development, the model is evaluated using metrics such as accuracy (Acc), Weighted Precision (W.P), Weighted Recall (W.R), Weighted F-score (W.F), Macro Precision (M.P), Macro Recall (M.R), and Macro F-score (M.F).

Due to highly imbalanced classes, this study employs both weighted and macro evaluations. SVM outperforms when it is applied with PCA

and Class Weights with an accuracy of 0.4386%, weighted precision of 0.6524%, weighted recall of 0.6459%, weighted F-score of 0.6435%, macro precision of 0.5191%, macro recall of 0.5224%, and the value of macro F-score of 0.5145%. The results of the SVM models are presented in Table 9.

Table 9: Results on the first four articles by applying SVM

	SVM	**SVM with PCA**	**SVM with Class Weights**	**SVM with PCA & Class Weights**
Acc	0.5205	0.4935	0.4817	0.4386
W.P	0.8786	0.8478	0.7194	0.6524
W.R	0.4591	0.4245	0.5709	0.6459
W.F	0.5534	0.5189	0.6299	0.6435
M.P	0.8247	0.7747	0.6144	0.5191
M.R	0.2667	0.2348	0.4220	0.5224
M.F	0.3738	0.3331	0.4917	0.5145

When SVM is implemented with PCA and Class Weights separately, it outperforms SVM using the combination of both PCA and Class Weights in terms of accuracy, achieving 0.4935% and 0.4817%, Weighted F-score with .5189% & 0.6299% and Macro F-score with 0.3331% & 0.4917% respectively. However, SVM with PCA performs very poorly while handling small classes. The performance of SVM with Class Weights is competitive with that of SVM using the combination of PCA and Class Weights in small classes as well. Nevertheless, the over-all results of SVM with the combination of PCA and Class Weights are better. The performance metrics of SVM with other combinations on the first four articles at SDG Goals are shown in Figure 20.

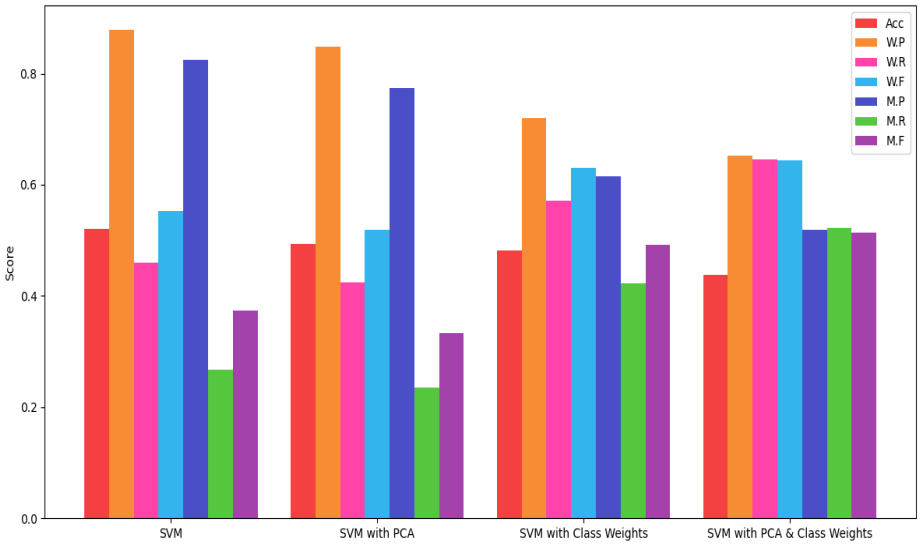

Figure 20: Performance metrics of SVM on first four articles at SDG Goals

Following the evaluation of the SVM models using the text data from the first four articles, the experimentation is extended by applying KNN and KNN with PCA. The results of KNN are better than KNN with PCA. KNN outperforms with 0.5151% in terms of accuracy, 0.7117% weighted precision, 0.5596% weighted recall, 0.6129% weighted F-score, 0.6274% macro precision, 0.3974% macro recall, and 0.4633% macro F-score. As discussed earlier, KNN does not Support Class Weights, so Class Weights are not incorporated with KNN. The results of the KNN models are presented in Table 10.

Table 10: Results on the first four articles with KNN and KNN + PCA

	KNN	**KNN with PCA**
Acc	0.5151	0.5075
W.P	0.7117	0.7025
W.R	0.5596	0.5533
W.F	0.6129	0.6069
M.P	0.6274	0.6089
M.R	0.3974	0.3854
M.F	0.4633	0.4521

The performance of KNN with PCA is also competitive compared to KNN, achieving 0.5075% accuracy, a weighted F-score of 0.6069%, and a macro F-score of 0.4521%. Figure 21 shows the performance metrics of KNN and KNN with PCA on the first four articles on SDG Goals.

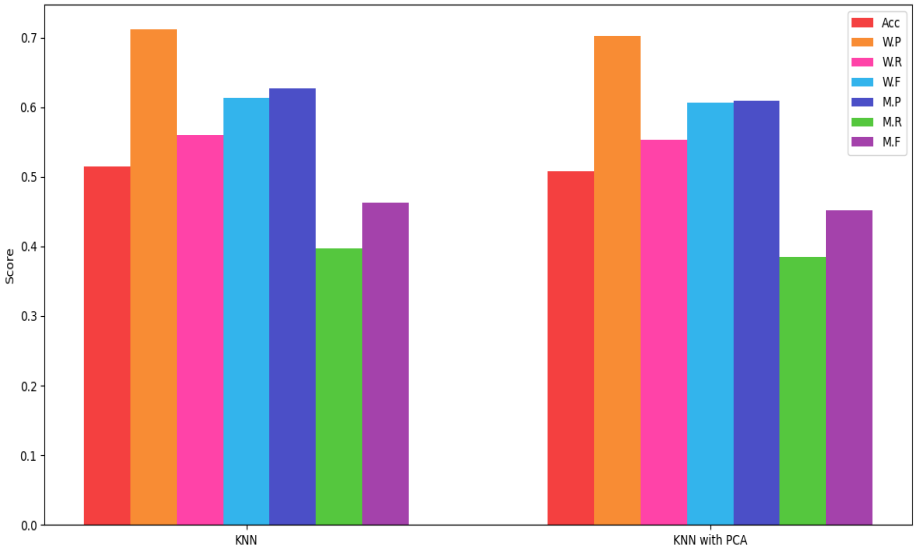

Figure 21: Performance metrics of KNN on first four articles at SDG Goals

4.4.1.2 All the articles

After experimenting with the text data from the first four articles, the study expands by taking the text data from all the articles as input variables to train and test the model. SVM is applied without PCA and Class Weights, with PCA and Class Weights both separately, and a combination of PCA and Class Weights. When the model is trained on the text data from all the articles, SVM demonstrates superior performance when implemented with Class Weights to handle imbalanced classes. SVM with Class Weights achieves an accuracy of 0.4806%, a weighted precision of 0.7057%, a weighted recall of 0.5823%, a weighted F-score of 0.6314%, a macro precision of 0.5981%, a macro recall of 0.4307%, and

a macro F-score of 0.4919%. The results of the SVM models are detailed in Table 11.

Table 11: Results of SVM on the text of all the articles as input variables

	SVM	SVM with PCA	SVM with Class Weights	SVM with PCA & Class Weights
Acc	0.5226	0.4946	0.4806	0.4267
W.P	0.8729	0.8714	0.7057	0.6415
W.R	0.4671	0.4205	0.5823	0.6464
W.F	0.5600	0.5137	0.6314	0.6378
M.P	0.8183	0.8241	0.5981	0.5069
M.R	0.2726	02336	0.4307	0.5212
M.F	0.3791	0.3310	0.4919	0.5074

SVM with PCA outperforms SVM with Class Weights in accuracy, achieving 0.4946% and in weighted and macro precision, with scores of 0.8714% and 0.8241%, weighted and Macro Recall of 0.4205% & 0.2336% and score of F-score both Weighted and Macro are 0.5137% & 0.3310% respectively. However, SVM's performance is very poor in the classification of small classes when implemented with PCA. Notably, SVM shows improved performance on small classes when using the combination of PCA and Class Weights, achieving a weighted F-score of 0.6378% and a macro F-score of 0.5074%. The performance metrics of SVM with other combinations on all the articles at SDG Goals are shown in Figure 22.

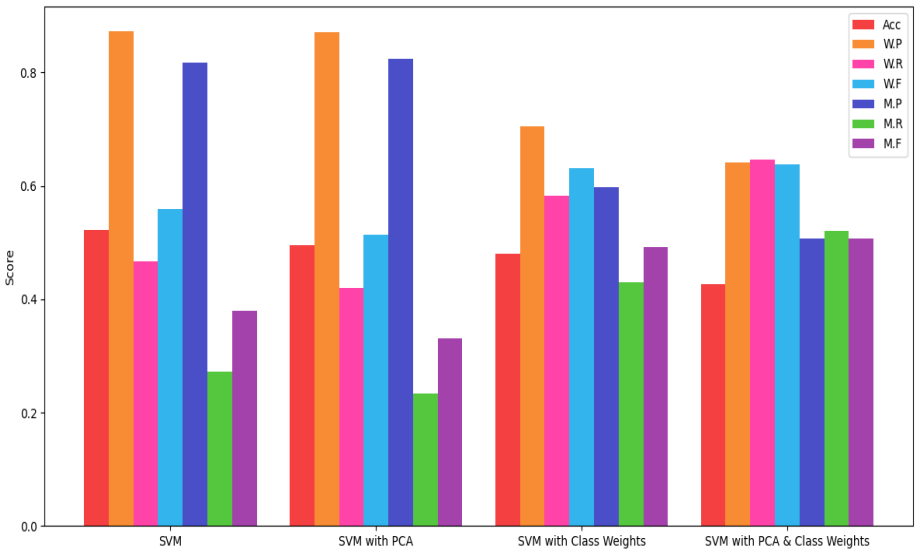

Figure 22: Performance metrics of SVM on all articles at SDG Goals

Following the SVM experimentation on the text data from all the articles, KNN is also tested. Initially, KNN is applied on its own, and subsequently, it is tested in combination with PCA. The experimentation reveals that KNN with PCA performs better, achieving an accuracy of 0.5097%, a weighted precision of 0.7038%, a weighted recall of 0.5533%, a weighted F-score of 0.6067%, a macro precision of 0.6108%, a macro recall of 0.3849%, and a macro F-score of 0.4515%. The results of the KNN models are detailed in Table 12.

Table 12: Results of KNN on the text of all articles as input variables

	KNN	**KNN with PCA**
Acc	0.5000	0.5097
W.P	0.6875	0.7038
W.R	0.5403	0.5533
W.F	0.5923	0.6067
M.P	0.5844	0.6108
M.R	0.3662	0.3849
M.F	0.4353	0.4515

The performance of KNN is also notable when implemented alone. KNN achieves 0.50% accuracy, with a weighted precision of 0.6875%, a weighted recall of 0.5403%, a weighted F-score of 0.5923%, a macro precision of 0.5844%, a macro recall of 0.3662%, and a macro F-score of 0.4353%. The performance metrics of KNN and KNN with PCA on all the articles at SDG Goals are shown in Figure 23.

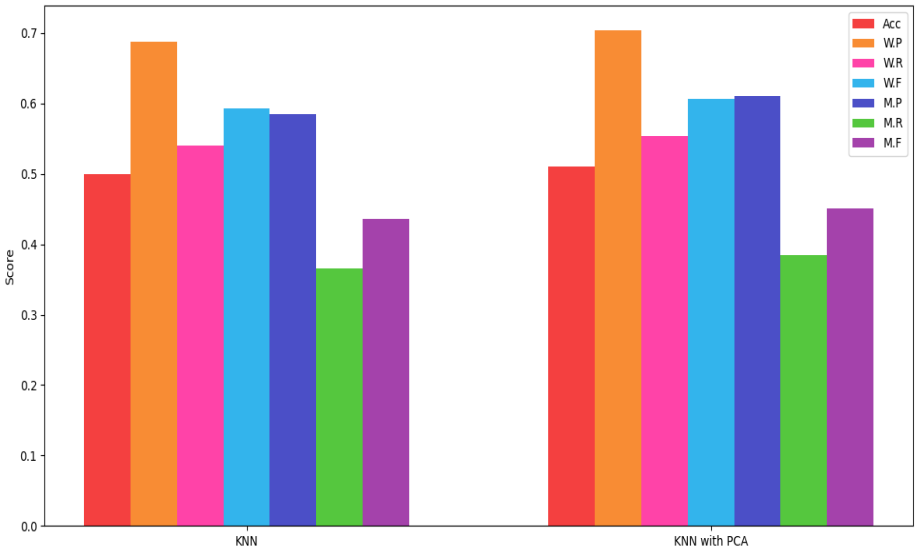

Figure 23: Performance metrics of KNN on all articles at SDG Goals

4.4.1.3 Preamble + first 4 articles

Building upon the experimentation, all articles were used as input variables. The study is further extended by incorporating the text data from the first four articles and the preambles as input variables.

The experiment begins with SVM, which is applied with the combination of PCA and Class Weights. When implemented with PCA and Class Weights, SVM demonstrates improved performance on smaller classes, as detailed in Table 13. The results show an accuracy of 0.5334%, a weighted precision of 0.7070%, a weighted recall of 0.7020%, a weighted F-score of 0.7004%, a macro precision of 0.5983%, a macro recall of 0.5738%, and a macro F-score of 0.5794%.

Table 13: Results of SVM on the text of first four articles + preambles as input variables

	SVM	SVM with PCA	SVM with Class Weights	SVM with PCA & Class Weights
Acc	0.5625	0.5506	0.5625	0.5334
W.P	0.8675	0.8673	0.7748	0.7070
W.R	0.5221	0.4989	0.6226	0.7020
W.F	0.6115	0.5917	0.6786	0.7004
M.P	0.8085	0.8038	0.7058	0.5983
M.R	0.3312	0.3119	0.4656	0.5738
M.F	0.4415	0.4228	0.5455	0.5794

The SVM, SVM with PCA, and SVM with Class Weights outperform the SVM with PCA and Class Weights in terms of accuracy with 0.5625%, 0.5506%, and 0.5625%, respectively. However, their performance in small classes is comparable to that of SVM with PCA, and their class weights are low. Figure 24 shows a visual representation of the performance of SVM, SVM with PCA, SVM with Class Weights, and SVM by combining both with SVM in the first four articles with preambles.

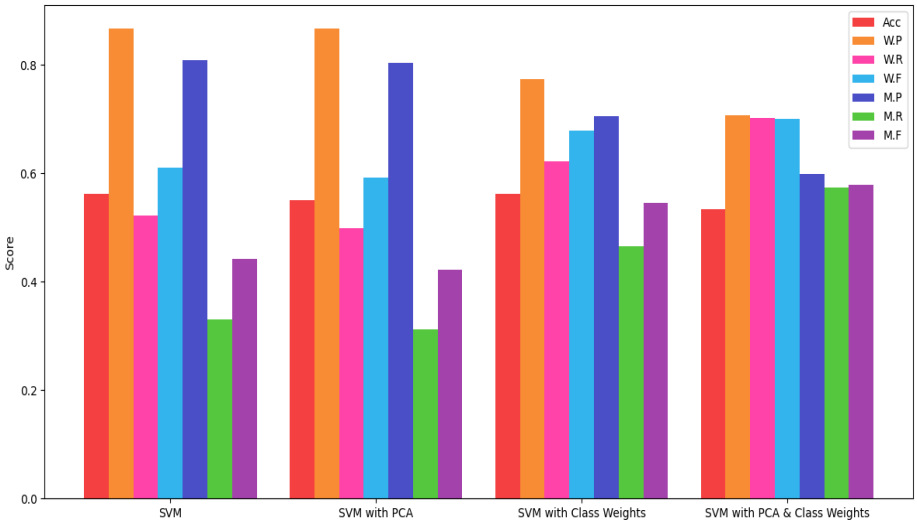

Figure 24: Performance metrics of SVM on first four articles with preambles at SDG Goals

Following the experimentation, using data from the first four articles and preambles as input variables, KNN was also tested. The results indicate that KNN performs better when combined with PCA than KNN alone. KNN with PCA achieves an accuracy of 0.5539%, a weighted precision of 0.7514%, a weighted recall of 0.5993%, a weighted F-score of 0.6499%, a macro precision of 0.6714%, a macro recall of 0.4262%, and a macro F-score of 0.4984%. The results of the KNN models are detailed in Table 14.

Table 14: Results of KNN on the text of first four articles + preambles as input variables

	KNN	**KNN with PCA**
Acc	0.5377	0.5539
W.P	0.7199	0.7514
W.R	0.6067	0.5993
W.F	0.6461	0.6499
M.P	0.6409	0.6714
M.R	0.4385	0.4262
M.F	0.5004	0.4984

KNN's performance is comparable to that of KNN with PCA. KNN achieves 0.5377% accuracy, a weighted precision of 0.7199%, a weighted recall of 0.6461%, a weighted F-score of 0.6409%, a macro precision of 0.4385%, and a macro F-score of 0.5004%. Figure 25 shows a visual representation of the performance of KNN and KNN with PCA on the first four articles with preambles at SDG Goals.

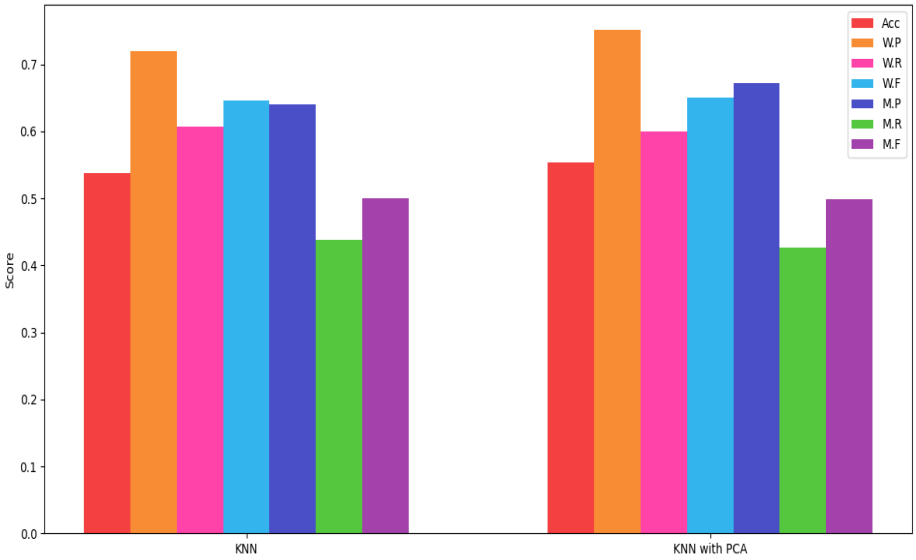

Figure 25: Performance metrics of KNN on first four articles with preambles at SDG Goals

4.4.1.4 Preamble + all the articles

After experimenting with the text data from all the preambles and the first four articles, the study is further extended by incorporating the text from all articles and all preambles as input variables.

The results reveal that SVM performs better when implemented with Class Weights compared to SVM with PCA or the combination of PCA and Class Weights. The best results achieved by SVM on all articles and preambles as input variables with an accuracy of 0.5657%, a weighted precision of 0.7715%, a weighted recall of 0.6209%, a weighted F-score

of 0.6764%, a macro precision of 0.7043%, a macro recall of 0.4623%, and a macro F-score of 0.5420%. The detailed results of the SVM models are presented in Table 15.

Table 15: Results of SVM on the text of all the articles + preambles as input variables

	SVM	SVM with PCA	SVM with Class Weights	SVM with PCA & Class Weights
Acc	0.5496	0.5496	0.5657	0.5474
W.P	0.8634	0.8629	0.7715	0.8635
W.R	0.4977	0.4960	0.6209	0.4977
W.F	0.5893	0.5867	0.6764	0.5887
M.P	0.7976	0.7959	0.7043	0.7972
M.R	0.3095	0.3067	0.4623	0.3086
M.F	0.4203	0.4164	0.5420	0.4187

The performance of SVM and SVM with either PCA or Class Weights separately outperforms SVM with the combination of PCA and Class Weights in both weighted and macro precision. However, the overall performance of SVM is better when using the combination of PCA and Class Weights. Figure 26 shows the visual representation of the performance of SVM, SVM with PCA, SVM with Class Weights, and SVM with PCA and Class Weights on text of all the articles with preambles.

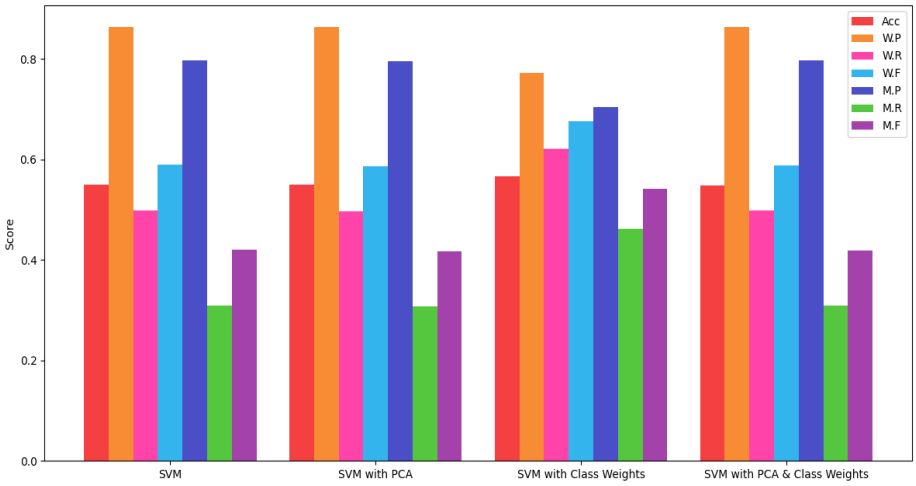

Figure 26: Performance metrics of SVM on all the articles with preambles at SDG Goals

Following the experimentation on all articles and preambles using SVM, KNN is also applied with the same input variables. The results indicate that KNN alone and KNN with PCA perform approximately equally; however, KNN alone demonstrates better classification for classes with fewer occurrences in the dataset. The detailed results are presented in Table 16.

Table 16: Results of KNN on the text of all the articles + preambles as input variables

	KNN	**KNN with PCA**
Acc	0.5453	0.5539
W.P	0.7244	0.7365
W.R	0.6044	0.5936
W.F	0.6461	0.6413
M.P	0.6395	0.6488
M.R	0.4319	0.4133
M.F	0.4953	0.4859

KNN achieves an accuracy of 0.5453%, while KNN with PCA achieves an accuracy of 0.5539%. The weighted precision values are 0.7244% and 0.7365%, respectively. The weighted recall values are 0.6044% and 0.5936%, and the weighted F-scores are 0.6461% and 0.6413%. The macro precision, recall, and F-score for KNN are 0.6395%, 0.6488%, and 0.4319%, respectively, while for KNN with PCA, they are 0.4133%, 0.4953%, and 0.4859%. The visual representation of the performance of KNN and KNN with PCA on all the articles with preambles at SDG Goals is shown in Figure 27.

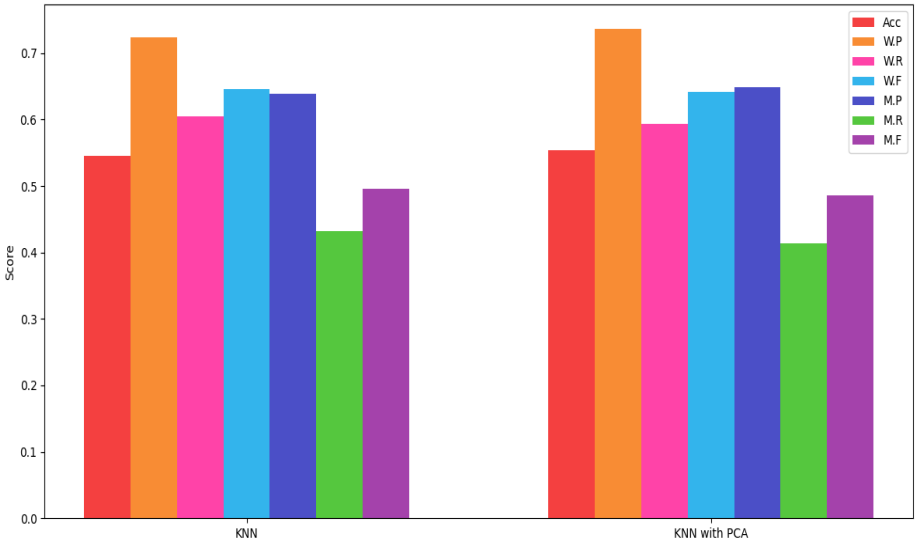

Figure 27: Performance metrics of KNN on all the articles with preambles at SDG Goals

4.4.2 Classification of SDGs Targets

As discussed above, the SDGs consist of 17 Goals, each with several Targets designed to achieve specific objectives. In total, there are 169 Targets associated with these 17 SDGs Goals.

Building on the classification of SDG Goals using SVM and KNN, this study further extends by classifying SDGs Targets through Machine Learning algorithms. To link the actions of the EU with the SDGs Targets. For this purpose, the experimentation is structured into four different scenarios: first, using the text of the first four articles as input

variables; second, utilizing the text data from all articles; third, combining the data of the first four articles with their preambles; and fourth, including the data of all articles along with their preambles to develop a model for the classification of SDGs Targets. Rest the details in the below sections.

4.4.2.1 First 4 articles

The first four articles were taken as input variables to develop a classification model for SDGs Targets. SVM was implemented without PCA and Class Weights, SVM with PCA, SVM with Class Weights, and by combining both Class Weights & PCA. Due to the large number of classes and the imbalanced class dataset, the results were not optimal. However, the SVM performed better compared to the other combinations, as shown in Table 17. SVM performs with an accuracy of 0.4538%, weighted precision of 0.7873%, weighted recall of 0.3461%, weighted F-score of 0.4337%, macro precision of 0.4567%, macro recall of 0.1476%, and macro F-score of 0.1988%.

Table 17: Results on the first four articles by applying SVM

	SVM	**SVM with PCA**	**SVM with Class Weights**	**SVM with PCA & Class Weights**
Acc	0.4538	0.4312	0.3495	0.3634
W.P	0.7873	0.7873	0.7873	0.7873
W.R	0.3461	0.3461	0.3461	0.3461
W.F	0.4337	0.4337	0.4337	0.4337
M.P	0.4567	0.4567	0.4567	0.4567
M.R	0.1476	0.1476	0.1476	0.1476
M.F	0.1988	0.1988	0.1988	0.1988

As discussed above, SVM outperforms when implemented without PCA and Class Weights. However, the accuracy of the results of SVM with PCA is relatively close. When evaluated with other evaluation measures, the performance of SVM with PCA is the same as the SVM's. Figure 28 shows a visual representation of the performance of SVM with different combinations of the first four articles at the SDGs classification as the Target level.

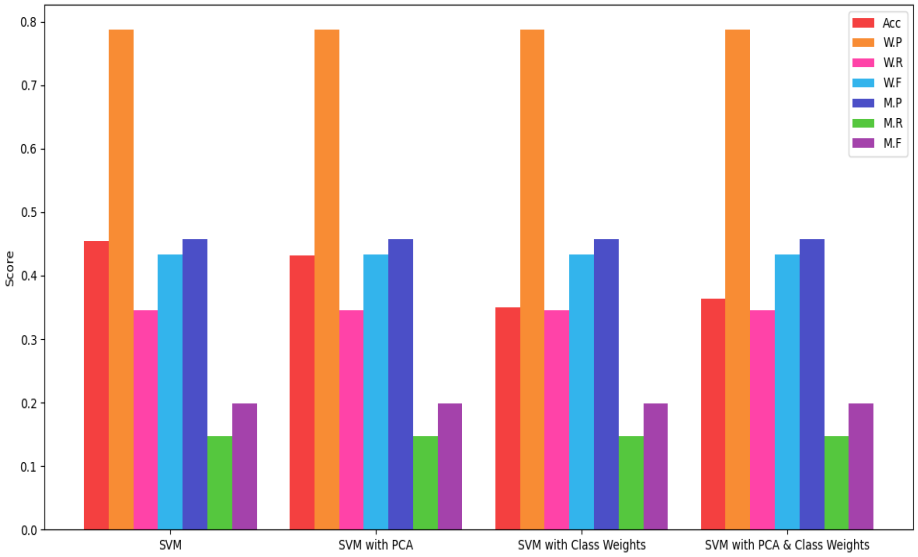

Figure 28: Performance metrics of SVM on first four articles at SDGs Targets

After obtaining the results from SVM, KNN was also implemented using the text from the first four articles as input, along with a combination of PCA. Class Weights were not included because KNN does not support Class Weights. The results of KNN by incorporating PCA were better than the results of KNN alone. KNN with PCA achieved an accuracy of 0.4763%, weighted precision of 0.6800%, weighted recall of 0.4475%, weighted F-score of 0.5147%, macro precision of 0.4031%, macro recall of 0.2107%, and macro F-score of 0.2557%. The results of SVM on the Target level are shown in Table 18.

Table 18: Results on the first four articles by applying KNN

	KNN	KNN with PCA
Acc	0.4806	0.4763
W.P	0.5758	0.6800
W.R	0.4687	0.4475
W.F	0.5041	0.5147
M.P	0.3126	0.4031
M.R	0.2303	0.2107
M.F	0.2504	0.2557

The KNN results are not as good as those of the KNN with PCA; however, they are promising when implemented using the first four articles as input variables, achieving an accuracy of 48.06%. Figure 29 shows a visual representation of the performance of KNN and KNN with PCA in the first four articles of the SDGs Target-level classification.

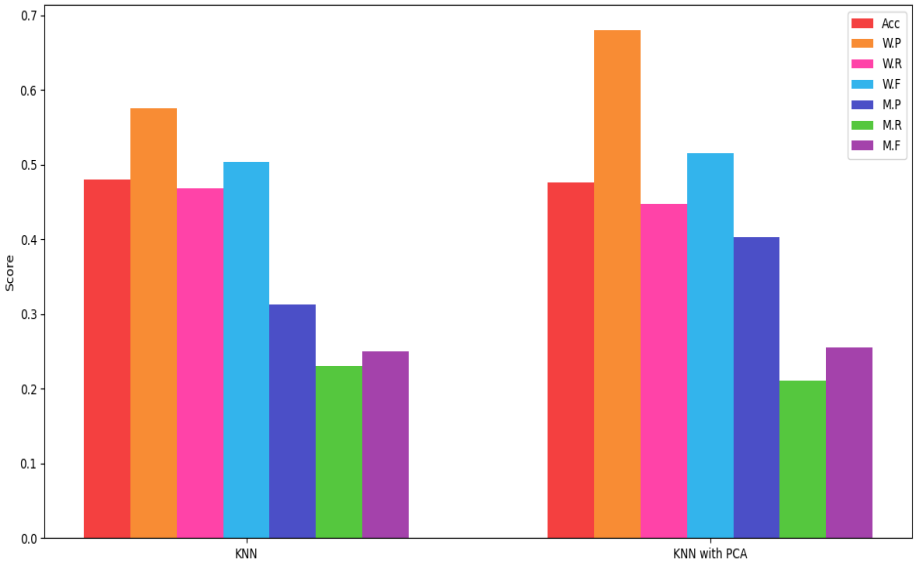

Figure 29: Performance metrics of KNN on first four articles at SDGs Targets

4.4.2.2 All the articles

After conducting the classification results by taking the first four articles as input, the results were extended by the text data from all the articles as input, and SVM was implemented for the classification of SDGs Targets. The SVM was applied alone, with PCA, with Class Weights, and with a combination of PCA and Class Weights. SVM with Class Weights performed well on smaller classes, achieving an accuracy of 0.3613%, weighted precision of 0.6501%, weighted recall of 0.4387%, weighted F-score of 0.5041%, macro precision of 0.3821%, macro recall of 0.2218%, and macro F-score of 0.2589%. The results of SVM on all articles are shown in Table 19.

Table 19: Results of SVM on the text of all the articles as input variables on SDGs Targets Classification

	SVM	**SVM with PCA**	**SVM with Class Weights**	**SVM with PCA & Class Weights**
Acc	0.4527	0.4323	0.3613	0.3323
W.P	0.7882	0.7812	0.6501	0.5760
W.R	0.3535	0.3157	0.4387	0.4673
W.F	0.4407	0.4032	0.5041	0.5017
M.P	0.4727	0.4485	0.3821	0.3035
M.R	0.1529	0.1345	0.2218	0.2299
M.F	0.2052	0.1825	0.2589	0.2446

The performance of SVM with PCA and Class Weights is suitable for small classes, but in terms of accuracy, SVM and SVM with PCA outperform by achieving 0.4527% and 0.4323% accuracy, respectively. The values of weighted precision and macro precision are also notable. Figure 30 shows a visual representation of the performance of SVM with

their different combinations on all the articles at the SDGs classification at the Target level.

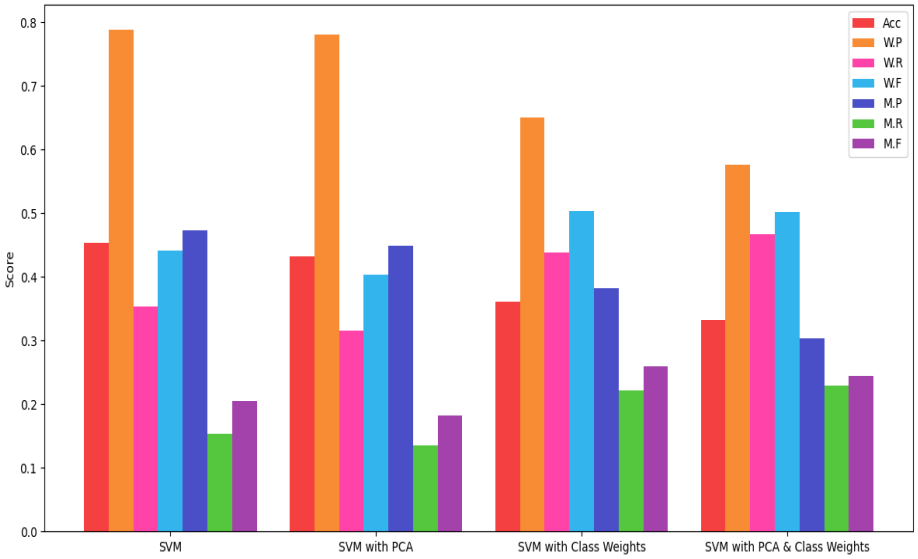

Figure 30: Performance of SVM on all articles at SDGs Targets

For the classification of SDGs Targets using all the articles' data as input, KNN performed similarly by incorporating PCA and without PCA as well. However, the results with PCA were slightly better, achieving an accuracy of 0.4796%, weighted precision of 0.6758%, weighted recall of 0.4452%, weighted F-score of 0.5102%, macro precision of 0.4045%, macro recall of 0.2101%, and macro F-score of 0.2553%. The remaining results of KNN on SDGs Targets using all the articles as input are presented in Table 20.

Table 20: Results of KNN on the text of all the articles as input variables on SDGs Targets Classification

	KNN	**KNN with PCA**
Acc	0.4720	0.4796
W.P	0.5907	0.6758
W.R	0.4567	0.4452
W.F	0.5001	0.5102

M.P	0.3294	0.4045
M.R	0.2290	0.2101
M.F	0.2532	0.2553

KNN without incorporating PCA performs with 0.4720% accuracy, 0.5907% weighted precision, 0.4567% weighted recall, 0.5001% weighted F-score, 0.3294% macro precision, 0.2290% macro recall, and a weighted F-score of 0.2532%. The visual representation of the performance of KNN and KNN with PCA on all the articles at the SDGs Target level classification is shown in Figure 31.

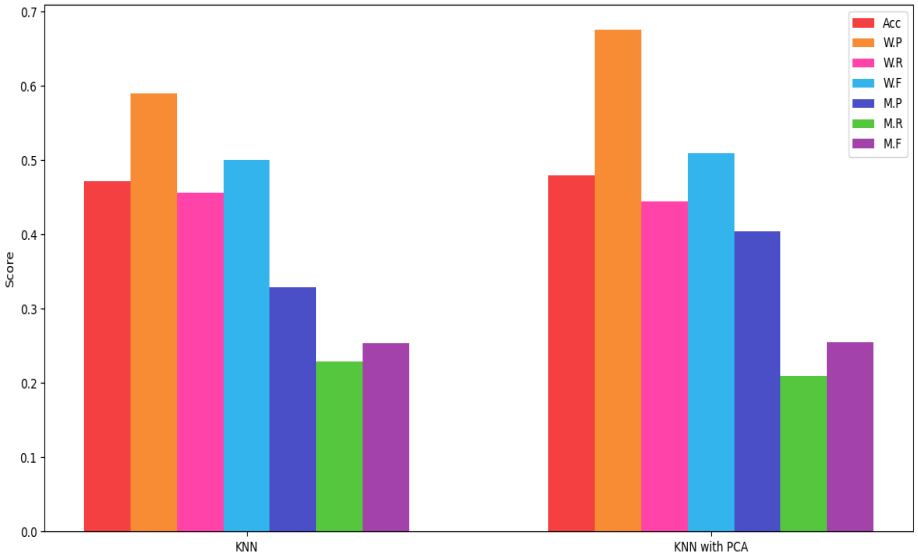

Figure 31: Performance metrics of KNN on all articles at SDGs Targets

4.4.2.3 Preamble + first 4 articles

After classifying the first four articles and all articles, the classifications were performed by taking the text from the first four articles along with the preambles text. Initially, the SVM algorithm was applied alone, followed by its combination with PCA and class weighting techniques, both individually and combined. The combination of SVM with PCA

and Class Weights yielded the best performance, achieving an accuracy of 51.94%, a weighted precision score of 72.72%, a weighted recall of 49.35%, a weighted F-score of 56.40%, a macro precision of 44.43%, a macro recall of 24.84%, and a macro F-score of 29.44%. The results of the SVM classification on four articles with preambles based on SDGs Targets are presented in Table 21.

Table 21: Results of SVM on the text of first four articles with preambles as input variables on SDGs Targets Classification

	SVM	**SVM with PCA**	**SVM with Class Weights**	**SVM with PCA & Class Weights**
Acc	0.5097	0.4860	0.4624	0.5194
W.P	0.7873	0.7873	0.6326	0.7272
W.R	0.3461	0.3461	0.5396	0.4935
W.F	0.4337	0.4337	0.5692	0.5640
M.P	0.4567	0.4567	0.3806	0.4443
M.R	0.1476	0.1476	0.2937	0.2484
M.F	0.1988	0.1988	0.3098	0.2944

The results of SVM with Class Weights are relatively close to those of SVM with both Class Weights and PCA. However, SVM with Class Weights outperforms SVM with PCA and Class Weights, achieving a weighted F-score of 0.5692% and a macro F-score of 0.3098%. Nevertheless, the overall results of SVM with PCA and Class Weights are better. The visual representation of the performance of SVM with their

different combinations by taking the first four articles with the preambles text as input variables at the SDGs classification at the Target level is shown in Figure 32.

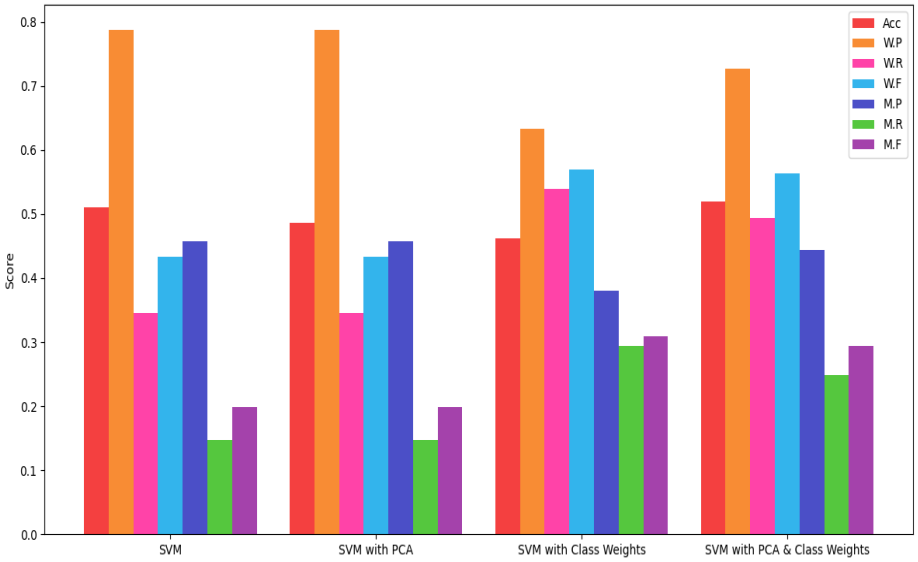

Figure 32: Performance metrics of SVM on first four articles with preambles at SDGs Targets

After analyzing the results of the SVM algorithm on the first four articles with preamble text as input variables, the KNN algorithm was also applied. Similar to SVM, KNN was tested with and without the PCA. KNN performed slightly better when combined with PCA, achieving an accuracy of 51.94%, a weighted precision score of 68.51%, a weighted recall of 44.06%, a weighted F-score of 50.84%, a macro precision of 41.22%, a macro recall of 20.62%, and a macro F-score of 25.27%. The results of the KNN classification on the text of the first four articles with preambles, based on SDGs Targets, are presented in Table 22.

Table 22: Results of KNN on the text of first four articles with preambles as input variables on SDGs Targets Classification

	KNN	**KNN with PCA**
Acc	0.5151	0.5194
W.P	0.5777	0.6851

W.R	0.5037	0.4406
W.F	0.5255	0.5084
M.P	0.4122	0.4122
M.R	0.2062	0.2062
M.F	0.2527	0.2527

The results of KNN with PCA are comparatively better, but the performance of KNN is also very similar. KNN performs similarly to KNN with PCA in terms of macro precision, recall, and F-score, with values of 0.4122%, 0.2062%, and 0.2527%, respectively. Figure 33 shows a visual representation of the performance of KNN and KNN with PCA by taking the first four articles with the preambles as input variables at the SDGs classification at the Target level.

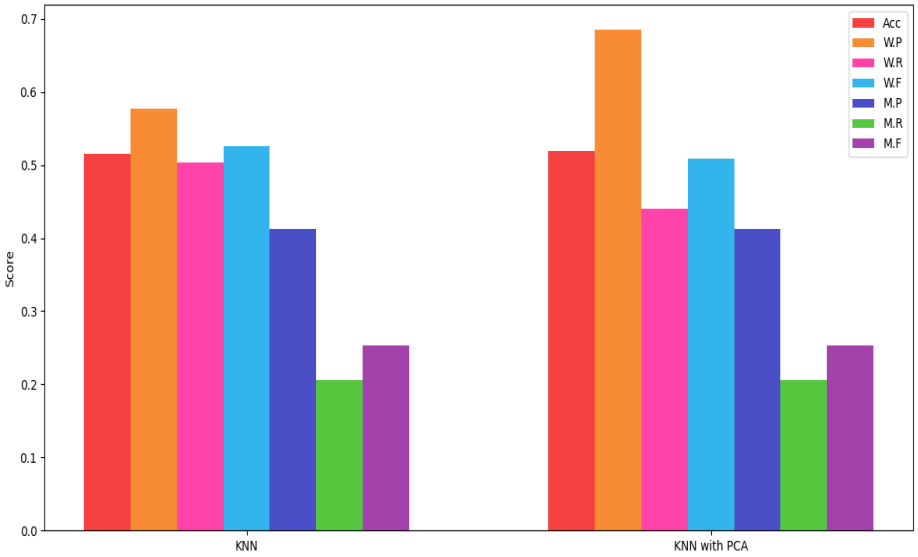

Figure 33: Performance metrics of KNN on first four articles with preambles at SDGs Targets

4.4.2.4 Preamble + all the articles

Finally, experiments were conducted using the text from all the articles, with the preambles included as input variables. The SVM algorithm was applied with various configurations: SVM alone, SVM with PCA, SVM with Class Weights, and SVM with the combination of PCA and Class Weights. Among these, SVM with PCA and Class Weights performed the best, effectively addressing the class imbalance in the dataset. This configuration achieved an accuracy of 46.56%, a weighted precision of 63.22%, a weighted recall of 53.5%, a weighted F-score of 56.6%, a macro precision of 37.38%, a macro recall of 28.99%, and a macro F-score of 30.61%. The results of the SVM classification of all the articles with preambles based on SDGs Targets are shown in Table 23.

Table 23: Results of SVM on the text of all articles with preambles as input variables on SDGs Targets Classification

	SVM	**SVM with PCA**	**SVM with Class Weights**	**SVM with PCA & Class Weights**
Acc	0.4882	0.4849	0.4860	0.4656
W.P	0.7535	0.7525	0.7527	0.6322
W.R	0.3608	0.3599	0.3608	0.535
W.F	0.4446	0.4435	0.4442	0.566
M.P	0.4420	0.4419	0.4417	0.3738
M.R	0.1564	0.1558	0.1563	0.2899
M.F	0.2080	0.2073	0.2078	0.3061

SVM, SVM with PCA and SVM with Class Weights outperform the SVM with the combination of PCA and Class Weights in terms of weighted precision with 0.7535%, 0.7525%, and 0.7527%, respectively. However, in the other evaluation measure, the SVM shows the best performance with PCA and Class Weights. The performance of SVM, SVM with PCA, and SVM with Class Weights are very close to each other in accuracy and weighted & macro F-score. The visual representation of the performance of SVM with their different combinations at the SDGs

classification at the Target level by taking all the articles with preambles text as an input variable is shown in Figure 34.

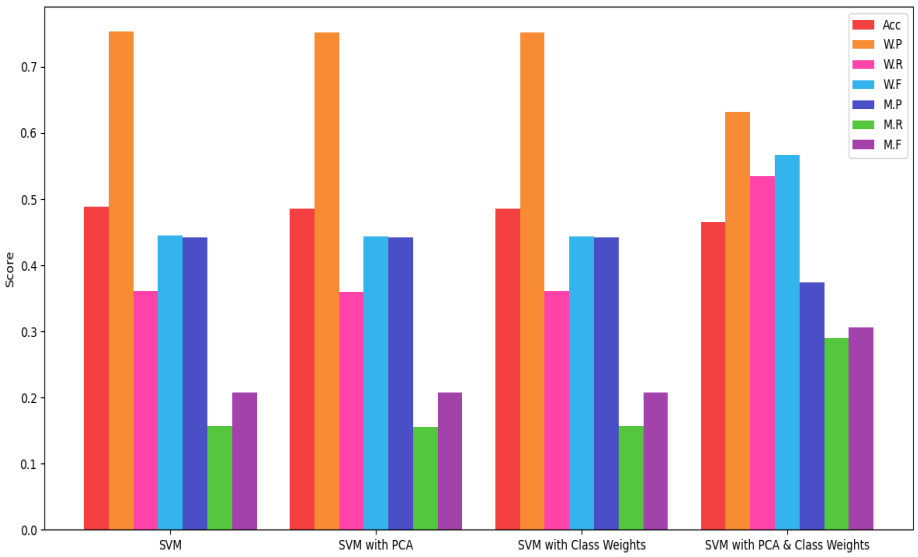

Figure 34: Performance metrics of SVM on all the articles with preambles at SDGs Targets

After conducting the experimentation by applying the SVM. The KNN algorithm was also applied to the text of all the articles with preambles, using TF-IDF and a combination of TF-IDF with PCA. The KNN algorithm performed similarly in both configurations, though slightly better when combined with PCA. With PCA, KNN achieved an accuracy of 51.83%, a weighted precision of 68.51%, a weighted recall of 44.06%, a weighted F-score of 50.84%, a macro precision of 41.22%, a macro recall of 20.62%, and a macro F-score of 25.27%. The results of the KNN classification on all the articles with preambles based on SDGs Targets are presented in Table 24.

Table 24: Results of KNN on the text of all the articles with preambles as input variables on SDGs Targets Classification

	KNN	**KNN with PCA**
Acc	0.5161	0.5183

W.P	0.6851	0.6851
W.R	0.4406	0.4406
W.F	0.5084	0.5084
M.P	0.4122	0.4122
M.R	0.2062	0.2062
M.F	0.2527	0.2527

By taking all the articles with the preambles, KNN achieves 0.5161% accuracy, 0.6851% weighted precision, 0.4406% weighted recall, 0.5084% weighted F-score, 0.4122% macro precision, 0.2062% macro recall, and 0.2527% macro F-score. Figure 35 shows the visual representation of the performance of KNN and KNN with PCA at the SDGs Target level classification by taking all the articles with preambles text as an input variable.

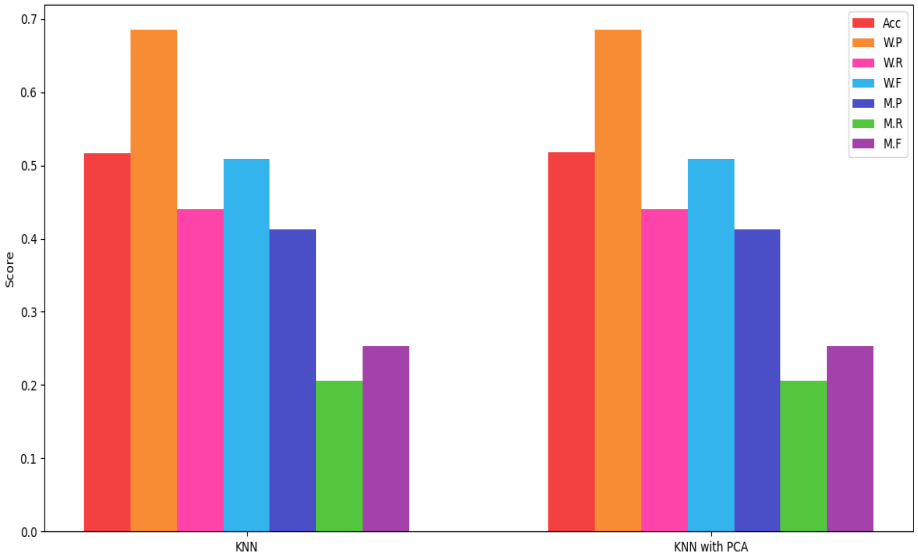

Figure 35: Performance metrics of KNN on all the articles with preambles at SDGs Targets

4.5 Saved the Best Model

Various Machine Learning (ML) algorithms (as discussed in section 4.3.2) were explored to identify the most effective model for classifying

Sustainable Development Goals (SDGs) at both the Goals and Targets levels. After experimentation, this study found that Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) performed significantly better than other algorithms.

After selecting SVM and KNN, these algorithms were applied using different combinations of articles and preambles text as input variables to evaluate their performance. Principal Component Analysis (PCA) is incorporated to enhance performance by reducing the text's dimensionality and addressing class imbalance issues; Class Weights are incorporated. PCA and Class Weights significantly impacted the models' performance, particularly for smaller classes.

Experiments were conducted separately on the SDGs, 17 Goals, and 169 Targets, utilizing the best preprocessing techniques' combinations (as discussed in section 4.2.1.2.1) with SVM and KNN. Following detailed experimentation and analysis, the best-performing models for both (Goals and Targets levels) were saved to link the actions taken by the EU in the EU legislative files with SDGs at the Goals and Targets levels. Details of the best-performing models are presented in the following sections.

4.5.1 Best Model for SDGs Classification at Goals Level

SVM and KNN were applied using different preprocessing techniques across four proposed experiments (see section 4.4.1). In the first experiment, the text of the first four articles is taken as input variables. In the second, the text of all articles is included. In the third, the text data from the first four articles with preambles is taken as input to develop the ML model. In the fourth, the text of all articles, along with the preambles, is taken as input variables.

Support Vector Machine (SVM) emerged as the best-performing model, particularly by classifying smaller classes effectively. SVM achieved the best performance when the text from the first four articles and preambles were used as input variables. Principal Component

Analysis (PCA) was employed for dimensionality reduction, and Class Weights was incorporated to address the class imbalance. This configuration significantly improved classification metrics, showcasing its capability to handle imbalanced datasets while maintaining high levels of accuracy and F-score.

In contrast, K-Nearest Neighbors (KNN) also delivered competitive results, particularly when PCA was applied. However, it did not match the nuanced classification abilities of SVM, especially when SVM was combined with PCA and Class Weights. Further experiments using the text from the first four articles with preambles reaffirmed SVM's superiority, particularly when PCA and Class Weights were used as the sole enhancement.

Based on these findings, the SVM model with PCA and Class Weights was identified as the best-performing model for classifying SDG Goals. This configuration was selected and saved as the optimal model for linking actions in EU legislation with the Sustainable Development Goals at the Goals level. Details of the best model for SDGs classification at the Target level are discussed in section 4.4.1.3. The confusion matrix of the best-performing model is shown in Figure 36.

4.5.2 Best Model for SDGs Classification at Targets Level

The best-performing model is achieved using the SVM algorithm based on the experiments conducted for classifying SDGs Targets. When implemented with Principal Component Analysis (PCA) and Class Weights, SVM demonstrates superior performance when all articles, including their preambles, are used as input variables. The incorporation of PCA reduces the dimensionality of the data, improving computational efficiency, while using Class Weights addresses the class imbalance issue, ensuring fair representation of minority classes.

This approach effectively captures the underlying patterns within the text, consistently outperforming other methods. It delivers higher accuracy, precision, recall, and F-score metrics across both weighted and macro evaluations. These results highlight the robustness of this

methodology in mapping EU legislative actions to SDGs Targets, providing a reliable framework for further analysis and decision-making.

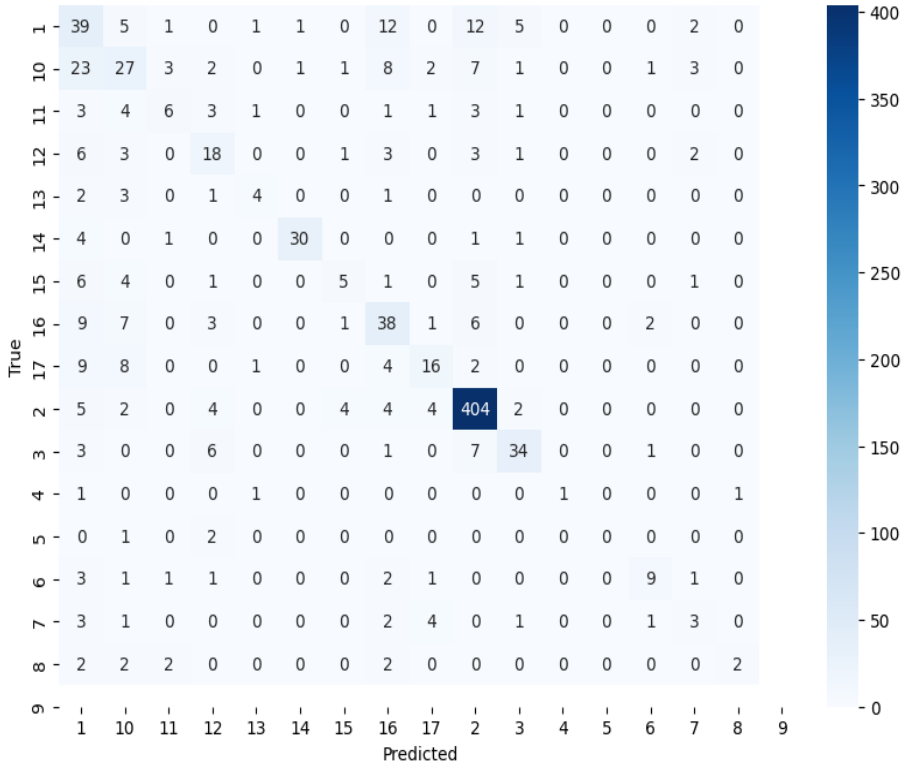

Figure 36: Confusion matrix of best performing model

In conclusion, the SVM model with PCA and Class Weights, utilizing the text from all the articles along with their preambles as input, is the most effective model for linking the EU's actions to the Sustainable Development Goals (SDGs) at the Target level within the legislative framework.

Chapter 5
Linking the Actions of EU Legislation with SDGs

5 Introduction

Chapter Four presents the details of the experimentation process and a comprehensive discussion of the results. Based on the analysis, the study identifies two optimal models for linking the actions of EU legislation with the SDGs at both Target and Goal levels. At the Goal level, the Support Vector Machine (SVM) model performs best when combined with Principal Component Analysis (PCA) for text dimensionality reduction and Class Weights to address class imbalance. This setup outperforms when it uses the first four articles' text and preamble text data as input variables. When implemented with PCA and Class Weights, the SVM model performs best at the Target level, incorporating the complete article text and preamble as input variables. These two top-performing models have been retained to link the actions in EU legislation with SDGs (Goals and Targets).

To link actions with the SDGs 17 Goals, 1,272 EU legislative files containing relevant *Definition*s were selected (the output files of independent annotation of *Definition*s discussed in section 3.10.3). The text from these files was extracted from the XML (LEOS dataset) files and saved in XLSX format. The text data is extracted from these 1272 files in 5 scenarios to link the SDGs at the Goals and Targets level. The scenarios are given below:

i. Text of First Four Articles
ii. Text of All the Articles
iii. Text of First Four Articles with Preambles
iv. Text of All the Articles with Preambles
v. All the Text of the Legislative File

Each extracted text was saved along with its corresponding CELEX identifier. This processed XLSX file was then provided to the saved SVM model to classify and link the actions found in the EU legislative files with the SDGs 17 Goals. Additional details on linking actions to EU legislation are provided in the subsequent sections.

5.1 SDGs Linking at the Goal Level

As discussed in the above section, the best-performing Machine Learning model for classifying SDG Goals is the Support Vector Machine (SVM) (see details in section 4.4.1.3). This model excels when combined with Principal Component Analysis (PCA) and Class Weights, using the first four articles along with the preamble text as input variable. The SVM model not only surpasses other models in performance but also effectively handles and classifies smaller classes. Due to its superior results and efficiency in managing smaller class distributions, this model has been preserved for linking EU legislation to the SDG; 17 Goals outlined by the United Nations. The linking of EU legislative files with SDG Goals is discussed in detail in the lower sections.

5.1.1 Text of First Four Articles

As discussed in the above section, the text of the first four articles of each file is extracted and saved in a XLSX file to link the SDGs at the Goal level. The saved file is provided to the trained model to predict the classes based on the training data. The saved SVM model links the text of each file with the relevant SDG Goals based on the training it received.

SDGs Goal 1 (No Poverty) is found in 240 files, and these 240 files are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is predicted in 110 files, and based on the predictions, these 110 files are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 248 files, and these 248 files are linked with SDGs Goal 3.

For SDGs Goal 4 (Quality Education), 63 files are linked based on the model's predictions. SDGs Goal 5 (Gender Equality) is associated with 8 files, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 22 files. Based on these associations, 8 files are linked to Goal 5, and 22 files are linked to Goal 6.

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 135 files, while SDGs Goal 8 (Decent Work and Economic Growth) is connected to 436 files based on the model's predictions. As per the model, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 352 files,

and SDGs Goal 10 (Reduced Inequality) is linked with 305 files. SDGs Goal 11 (Sustainable Cities and Communities) is associated with 120 files, and these 120 files are linked with SDGs Goal 11.The frequency of each SDGs found in 1,272 legislative files is shown in Figure 37.

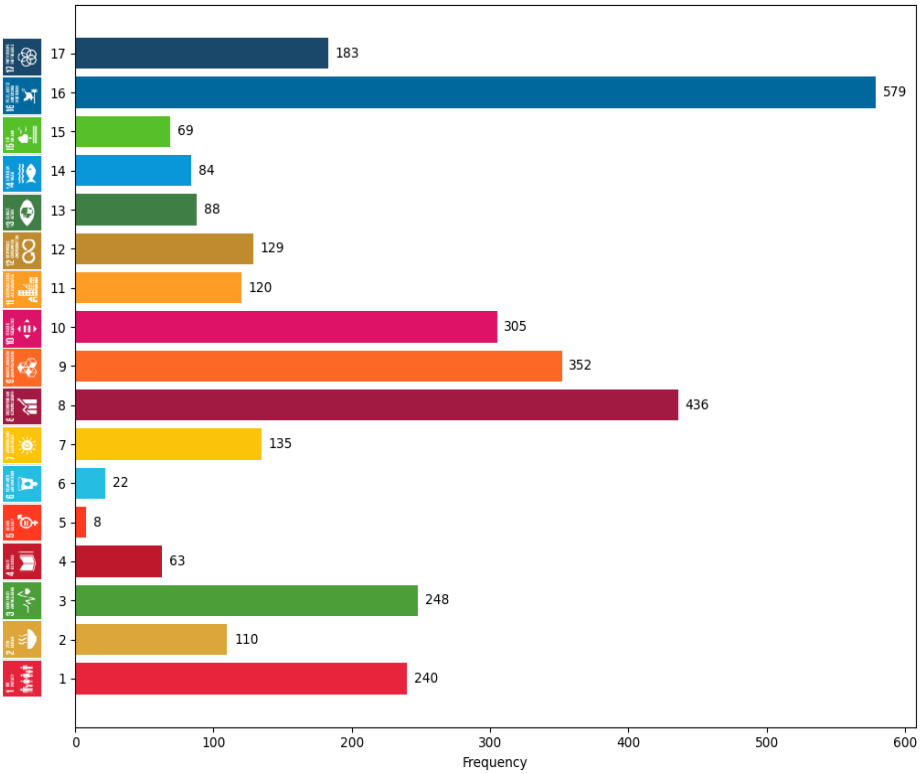

Figure 37: The frequency of each SDGs Goal found in 1,272 legislative files by taking the first four Articles as input

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 129 files, and SDGs Goal 13 (Climate Action) is also found in 88 files. As per the model's predictions, both SDGs Goal 12 and Goal 13 are linked to 129 and 88 files, respectively.

As predicted by the saved SVM model, 84 files are linked to SDGs Goal 14 (Life Below Water) and 69 to SDGs Goal 15 (Life on Land).

SDG Goal 16 (Peace, Justice, and Strong Institutions) is associated with 579 files, which are linked to SDG Goal 16. Finally, SDG Goal 17

(Partnerships for the Goals) is associated with 183 files, which are linked to SDG Goal 17.

5.1.2 Text of All the Articles

Linking SDG Goals with legislative files to identify the actions taken by the EU is also performed on the text of all articles. After extracting the text of all the articles, the saved SVM model provides the data to predict the actions found in the text and linked with SDG Goals.

SDGs Goal 1 (No Poverty) is found in 241 files, and these 241 files are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is predicted in 109 files, and based on the predictions, these 109 files are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 219 files, and these 219 files are linked with SDGs Goal 3.

For SDGs Goal 4 (Quality Education), 66 files are linked based on the model's predictions. SDGs Goal 5 (Gender Equality) is associated with 11 files, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 16 files. Based on these associations, 11 files are linked to Goal 5, and 16 files are linked to Goal 6.

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 138 files, while SDGs Goal 8 (Decent Work and Economic Growth) is connected to 420 files based on the model's predictions. As per the model, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 323 files, and SDGs Goal 10 (Reduced Inequality) is linked with 297 files. SDGs Goal 11 (Sustainable Cities and Communities) is associated with 103 files, and these 103 files are linked with SDGs Goal 11.

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 122 files, and SDGs Goal 13 (Climate Action) is also found in 93 files. As per the model's predictions, both SDGs Goal 12 and Goal 13 are linked to 122 and 93 files, respectively. As predicted by the saved SVM model, 88 files are linked to SDGs Goal 14 (Life Below Water) and 69 to SDGs Goal 15 (Life on Land).

SDGs Goal 16 (Peace, Justice, and Strong Institutions) is associated with 585 files, and these 585 files are linked with SDGs Goal 16. Finally, SDGs Goal 17 (Partnerships for the Goals) is associated with 193 files,

and these 193 files are linked with SDGs Goal 17. The frequency of each SDGs found in 1,272 legislative files by taking all articles' data as input is shown in Figure 38.

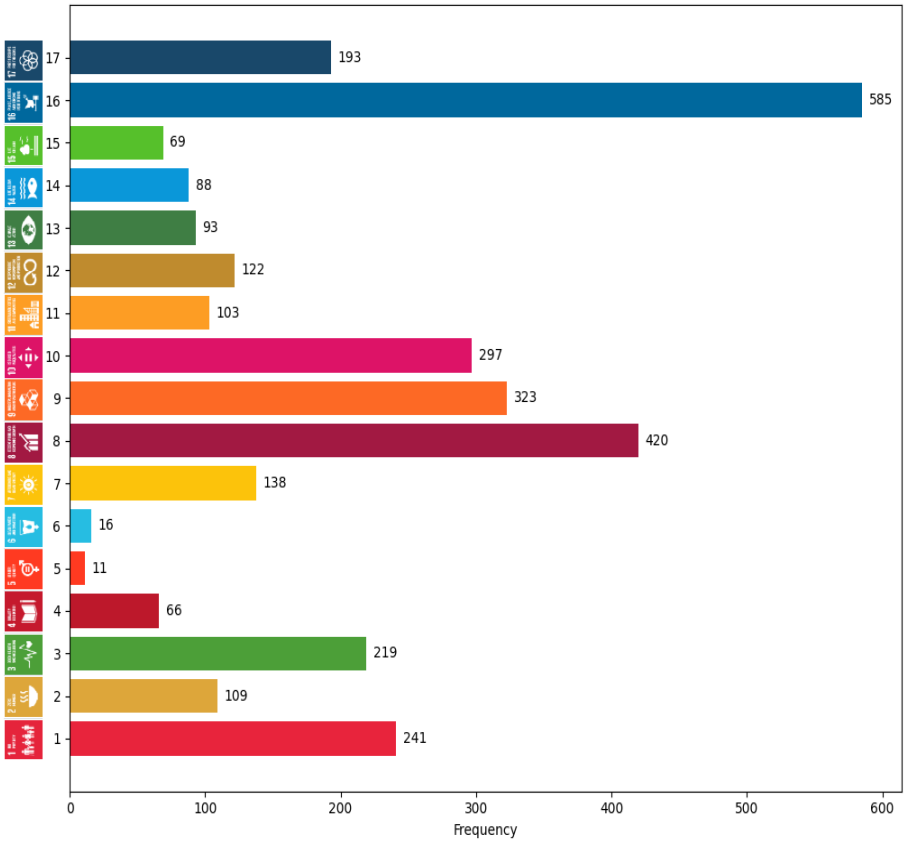

Figure 38: The frequency of each SDGs Goal found in 1,272 legislative files by taking all Articles' Text as input

5.1.3 Text of First Four Articles with Preambles

Linking SDG Goals with legislative files to identify the actions taken by the EU is also performed using the text of the first four articles with preambles. After extracting the text of the first four articles, the saved SVM model provides the data to predict the actions found in the text.

SDGs Goal 1 (No Poverty) is found in 299 files, and these 299 files are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is predicted in 150 files, and based on the predictions, these 150 files are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 332 files, and these 332 files are linked with SDGs Goal 3.

For SDGs Goal 4 (Quality Education), 87 files are linked based on the model's predictions. SDGs Goal 5 (Gender Equality) is associated with 14 files, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 40 files. Based on these associations, 14 files are linked to Goal 5, and 40 files are linked to Goal 6.The frequency of each SDGs found in 1,272 legislative files by taking the text of the first four articles with preambles as input is shown in Figure 39.

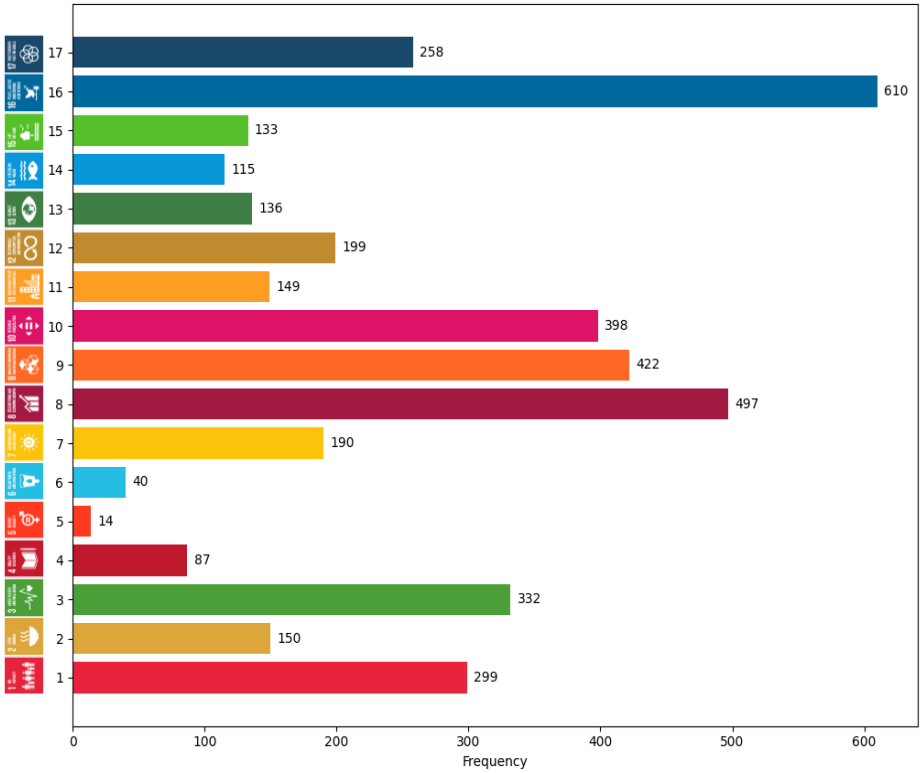

Figure 39: The frequency of each SDGs Goal found in 1,272 legislative files by taking the text of the first four Articles with preambles

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 190 files, while SDGs Goal 8 (Decent Work and Economic Growth) is connected

to 497 files based on the model's predictions. As per the model, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 422 files, and SDGs Goal 10 (Reduced Inequality) is linked with 398 files. SDGs Goal 11 (Sustainable Cities and Communities) is associated with 149 files, and these 149 files are linked with SDGs Goal 11.

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 199 files, and SDGs Goal 13 (Climate Action) is also found in 136 files. Both SDGs Goal 12 and Goal 13 are linked with 199 and 136 files, respectively, as per the model's predictions. With SDGs Goal 14 (Life Below Water), 115 files are linked, and with SDGs Goal 15 (Life on Land), 133 files are linked, as predicted by the saved SVM model.

SDGs Goal 16 (Peace, Justice, and Strong Institutions) is associated with 610 files, and these 610 files are linked with SDGs Goal 16. Finally, SDGs Goal 17 (Partnerships for the Goals) is associated with 258 files, and these 258 files are linked with SDGs Goal 17.

5.1.4 Text of All the Articles with Preambles

Linking SDG Goals with legislative files to identify the actions taken by the EU is also performed on the text of all the articles with preambles. After extracting the text of all the articles with the preambles, the saved SVM model provides the data to predict the actions found in the text.

SDGs Goal 1 (No Poverty) is found in 290 files, and these 290 files are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is predicted in 149 files, and based on the predictions, these 149 files are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 304 files, and these 304 files are linked with SDGs Goal 3.

For SDGs Goal 4 (Quality Education), 83 files are linked based on the model's predictions. SDGs Goal 5 (Gender Equality) is associated with 16 files, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 23 files. Based on these associations, 16 files are linked to Goal 5, and 23 files are linked to Goal 6.

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 184 files, while SDGs Goal 8 (Decent Work and Economic Growth) is connected to 484 files based on the model's predictions. As per the model, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 415 files, and SDGs Goal 10 (Reduced Inequality) is linked with 368 files. SDGs Goal 11 (Sustainable Cities and Communities) is associated with 134 files, and these 134 files are linked with SDGs Goal 11. Figure 40 shows the frequency of each SDG found in 1,272 legislative files, using the text of all the articles with preambles as input.

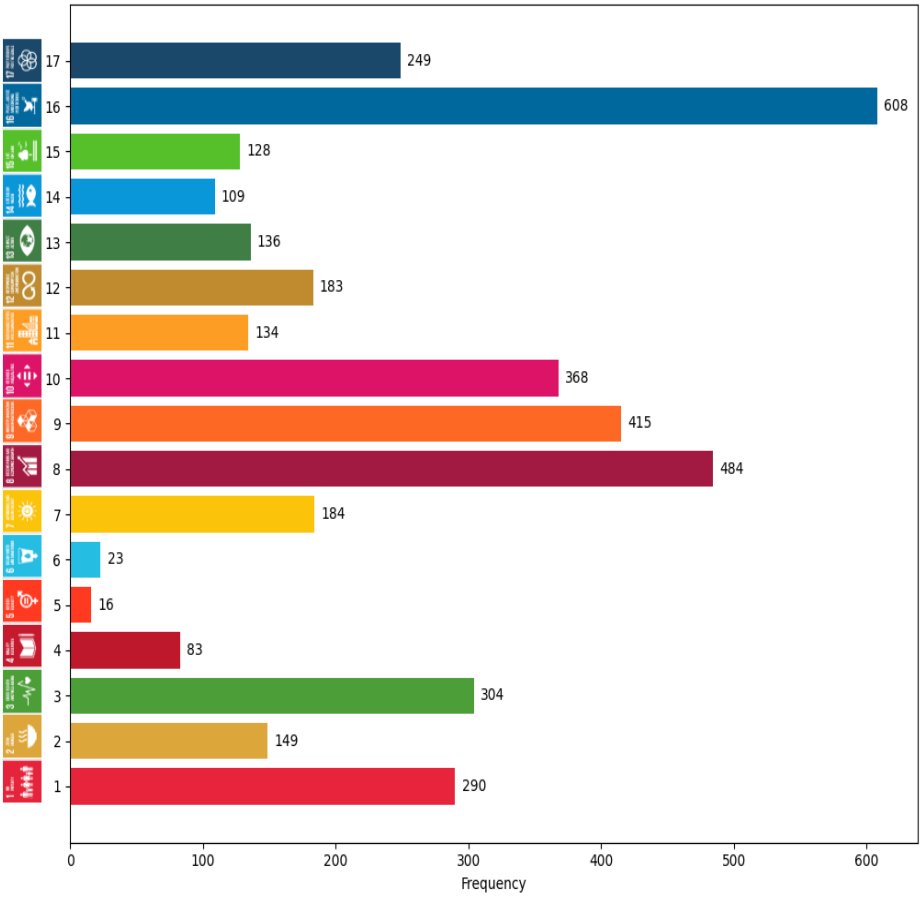

Figure 40: The frequency of each SDGs Goal found in 1,272 legislative files by taking the text of all the Articles with preambles as input

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 183 files, while SDGs Goal 13 (Climate Action) is found in 136

files. Both SDGs Goal 12 and Goal 13 are linked with 183 and 136 files, respectively, as per the model's predictions. With SDGs Goal 14 (Life Below Water), 109 files are linked, and with SDGs Goal 15 (Life on Land), 128 files are linked, as the saved SVM model predicted.

SDGs Goal 16 (Peace, Justice, and Strong Institutions) is associated with 608 files, which are linked to SDGs Goal 16. Finally, SDGs Goal 17 (Partnerships for the Goals) is associated with 249 files, which are linked to SDGs Goal 17.

5.1.5 All the Text of the File

Linking SDG Goals with legislative files to identify the actions taken by the EU is also carried out on the entire text of each file. After extracting the full text from all the files, the data is given to the SVM model to predict the actions described in the files.

SDGs Goal 1 (No Poverty) is found in 297 files, and these 297 files are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is predicted in 157 files, and based on the predictions, these 157 files are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 326 files, and these 326 files are linked with SDGs Goal 3.

For SDGs Goal 4 (Quality Education), 77 files are linked based on the model's predictions. SDGs Goal 5 (Gender Equality) is associated with 15 files, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 30 files. Based on these associations, 15 files are linked to Goal 5, and 30 files are linked to Goal 6.

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 217 files, while SDGs Goal 8 (Decent Work and Economic Growth) is connected to 497 files based on the model's predictions. As per the model, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 476 files, and SDGs Goal 10 (Reduced Inequality) is linked with 381 files. SDGs

Goal 11 (Sustainable Cities and Communities) is associated with 173 files, and these 173 files are linked with SDGs Goal 11.

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 193 files, and SDGs Goal 13 (Climate Action) is also found in 140 files. As per the model's predictions, both SDGs Goal 12 and Goal 13 are linked to 193 and 140 files, respectively. With SDGs Goal 14 (Life Below Water), 108 files are linked, and with SDGs Goal 15 (Life on Land), 135 files are linked, as the saved SVM model predicted. The frequency of each SDGs found in 1,272 legislative files by taking the entire text of each file as input is shown in Figure 41.

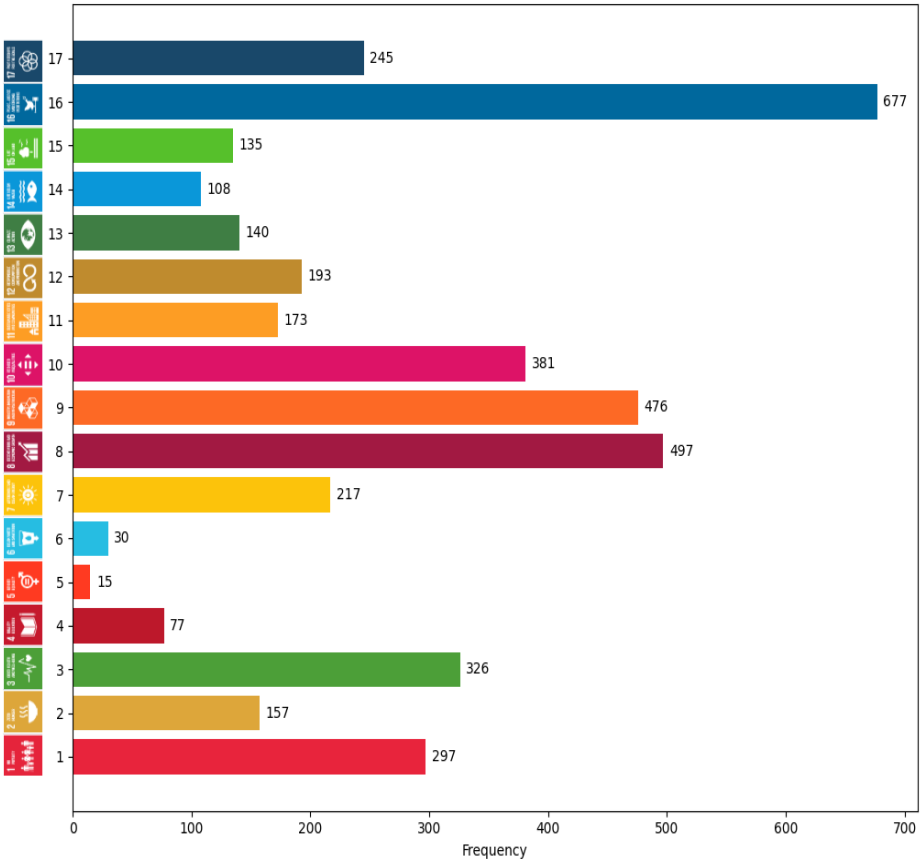

Figure 41: The frequency of each SDGs Goal found in 1,272 legislative files by taking the entire text of each file as input

SDGs Goal 16 (Peace, Justice, and Strong Institutions) is associated with 677 files, and these 677 files are linked with SDGs Goal 16. Finally,

SDGs Goal 17 (Partnerships for the Goals) is associated with 245 files, and these 245 files are linked with SDGs Goal 17.

5.2 SDGs Linking at the Target Level

The previous section presented linking actions identified in EU legislation to the SDG Goals. Building on this, the EU legislative documents were also connected to the SDGs at the Target level. For this purpose, the best-performing model, as discussed earlier in section 4.4.2.4, is the Support Vector Machine (SVM) combined with Principal Component Analysis (PCA) and Class Weights. This model utilizes the text of all articles' text and preambles as input variables to achieve optimal results. The linking of legislative files with SDGs Targets is done using 5 scenarios. A detailed discussion of linking EU legislative files with SDGs Targets is presented in lower sections.

5.2.1 Text of First Four Articles

Linking SDGs Targets with legislative files identifies the actions taken by the EU across 1,272 legislative files. First, the legislative files are linked with SDGs Targets with the text of the first four articles. This data is fed into the saved SVM model to predict the actions described in the files.

The model linked one file with Target 1.1. For Target 1.2, the number of linked files is 26. Target 1.3 is found in 77 files, which are linked with it. Sixty-six files are linked with Target 1.4, while 10 files are predicted and linked with Target 1.5. Target 2.1 is predicted in 55 files, and all these files are linked with it. Target 2.2 is found in 20 files, which are linked accordingly. The model links 46 files with Target 2.3. For Target 2.4, five files are linked. Additionally, 15 legislative files are linked with Target 2.5.

For Target 3.1, one file is linked. Twenty-two files are linked with Target 3.2, while Target 3.3 is linked with 10 files. A total of 113 files

are linked with Target 3.4 as per the model's predictions. For Target 3.5, 14 files are linked. Target 3.7 is linked with 12 files, and Target 3.8 is linked with 41 files. Finally, three files are linked with Target 3.9.

Five files are linked with Target 4.1, while two files are linked with Target 4.2. Target 4.3 is predicted in eight files, and 27 files are linked with Target 4.4. These files are linked based on the model's predictions. Target 4.6 is linked with three files. For Target 5.1, one file is linked, while 12 files are linked with Target 5.2. Seven files are associated with Target 5.3. One file is linked with Target 5.4.

Twelve files are linked with Target 6.1, while four files are linked with Target 6.2. Ten files are linked with Target 6.3 based on predictions, and one file is linked with Target 6.5. Forty-four files are linked with Target 7.1, while 33 files are linked with Target 7.2. For Target 7.3, 86 files are linked based on predictions.

For Target 8.1, 106 files are linked, while 73 files are linked with Target 8.2. Thirty-seven files are linked with Target 8.3, and two files are linked with Target 8.4. Twenty-two files are linked with Target 8.5, and one file is linked with Target 8.6. Four files are linked with Target 8.7, and 35 files are linked with Target 8.8.

Target 9.1 is found in 58 files, while Target 9.2 is found in five files. Seventy-seven files are linked with Target 9.3, and 25 files are linked with Target 9.4. For Target 9.5, 101 files are linked based on predictions. One file is linked with Target 10.1, and 12 files are linked with Target 10.2. A total of 124 files are linked with Target 10.3, while one file is linked with Target 10.4. Seven files are linked with Target 10.5, and 79 files are linked with Target 10.6.

Target 11.1 is found in 26 files. Seventy files are linked with Target 11.2, five files are linked with Target 11.4, six files are linked with Target 11.6, and one file is linked with Target 11.7. The frequency of each SDGs Target found in 1,272 legislative files by taking the text of the first four articles' as input is shown in Figure 42.

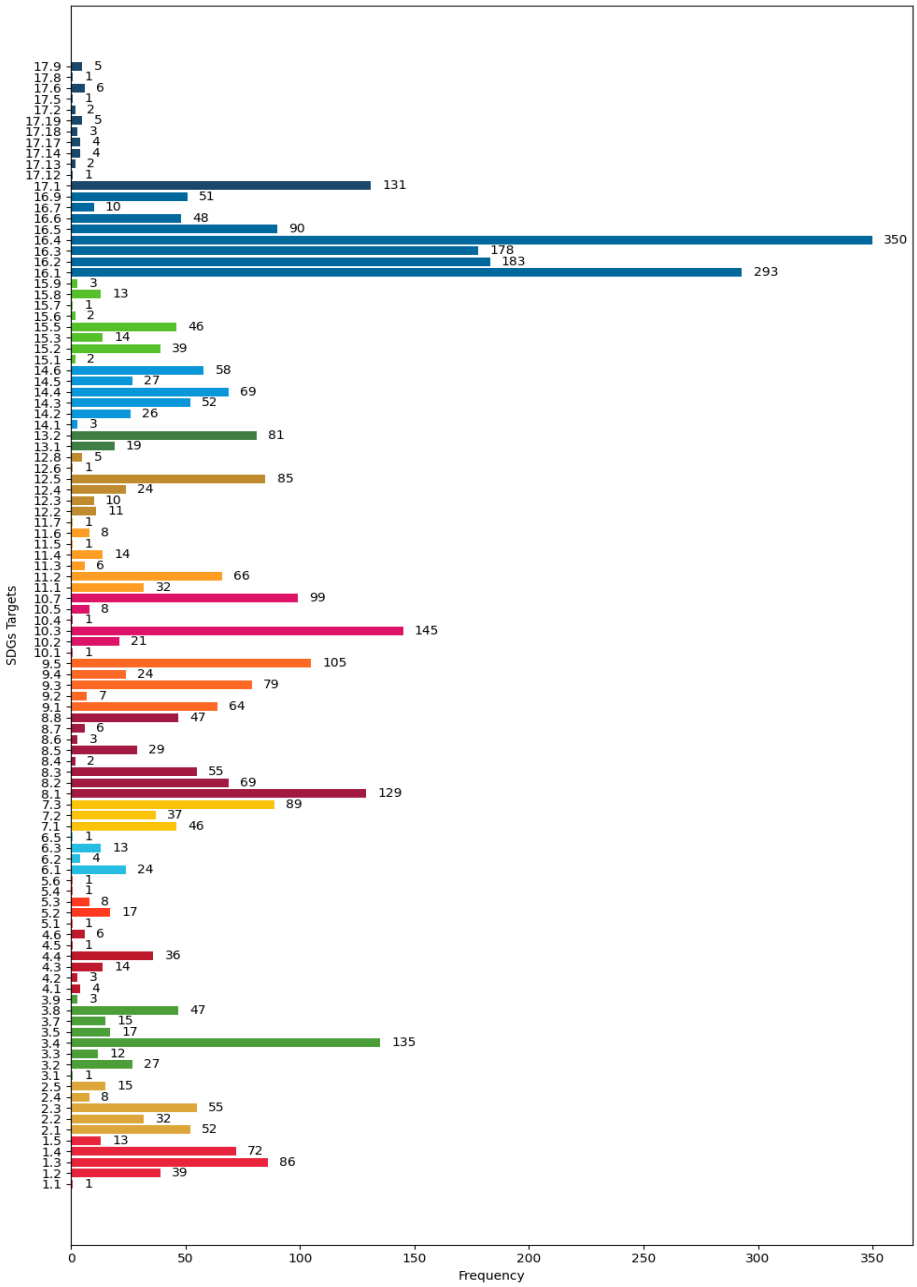

Figure 42: The frequency of each SDGs Target found in 1,272 legislative files by taking the text of the first four Articles as input

Nine files are linked with Target 12.2, while nine files are linked with Target 12.3. Twenty files are linked with Target 12.4, and 75 files are linked with Target 12.5. One file is linked with Target 12.6, and five files are linked with Target 12.8. For Target 13.1, 16 files are linked, while 72 files are linked with Target 13.2. Target 14.1 is linked with four files. Twenty-five files are linked with Target 14.2, 39 are linked with Target 14.3, 67 are linked with Target 14.4, 17 are linked with Target 14.5, and 47 are linked with Target 14.6.

Two files are linked to Target 15.1, 32 files are linked to Target 15.2, 11 files are linked to Target 15.3, 31 files are linked to Target 15.5, one file is linked to Target 15.7, eight files are linked to Target 15.8, and three files are linked to Target 15.9. For Target 16.1, 266 files are linked, while 150 files are linked with Target 16.2. Target 16.3 is linked with 150 files, and 316 files are linked with Target 16.4. Seventy-one files are linked with Target 16.5, while 35 files are linked with Target 16.6. Nine files are linked with Target 16.7, and 39 files are linked with Target 16.9.

For Target 17.1, 101 files are linked, while one file is linked with Target 17.2 and another one with Target 17.5. Six files are linked with Target 17.6, and one file is linked with Target 17.8. Four files are linked with Target 17.9, and one file is linked with Target 17.12. One file is linked with Target 17.13, and three files are linked with Target 17.14. Finally, four files are linked with Target 17.17, and two files are linked with Target 17.19.

5.2.2 Text of All the Articles

Linking SDGs Targets with legislative files identifies the actions taken by the EU across 1,272 legislative files with the text of all articles. This data is fed into the saved SVM model to predict the actions described in the files.

The model linked 1 file with Target 1.1. For Target 1.2, the number of linked files is 39. Target 1.3 is found in 86 files which are linked with it. The 72 files are linked with Target 1.4, while 13 files are predicted and linked with Target 1.5. Target 2.1 is predicted in 52 files, and all these files are linked with it. Target 2.2 is found in 32 files, which are linked accordingly. The model links 55 files with Target 2.3. For Target 2.4, 8

files are linked. Additionally, 15 legislative files are linked with Target 2.5.

For Target 3.1, 1 file is linked. 27 files are linked with Target 3.2, while 12 files are linked with Target 3.3. As per the model's predictions, 135 files are linked with Target 3.4. For Target 3.5, 17 files are linked. 15 files are linked with Target 3.7, and 47 files are linked with Target 3.8. Finally, 3 files are linked with Target 3.9.

4 files are linked with Target 4.1, while 3 files are linked with Target 4.2. Target 4.3 is predicted in 14 files, and 36 files are linked with Target 4.4. These files are linked based on the model’s predictions, and 1 file is linked with Target 4.5. Target 4.6 is linked with 6 files. For Target 5.1, 1 file is linked, while 17 files are linked with Target 5.2. 8 files are associated with Target 5.3. 1 file is linked with Target 5.4, and 1 file is linked with Target 5.6.

24 files are linked with Target 6.1, while 4 files are linked with Target 6.2. 13 files are linked with Target 6.3 based on predictions, and 1 file is linked with Target 6.5. 46 files are linked with Target 7.1, while 37 files are linked with Target 7.2. For Target 7.3, 89 files are linked based on predictions.

For Target 8.1, 129 files are linked, while 69 files are linked with Target 8.2. 55 files are linked with Target 8.3, and 2 files are linked with Target 8.4. 29 files are linked with Target 8.5, and 3 files are linked with Target 8.6. 6 files are linked with Target 8.7, and 47 files are linked with Target 8.8.

Target 9.1 is found in 64 files, while Target 9.2 is found in 7 files. 79 files are linked with Target 9.3, and 24 files are linked with Target 9.4. For Target 9.5, 105 files are linked based on predictions. 1 file is linked with Target 10.1, and 21 files are linked with Target 10.2. A total of 145 files are linked with Target 10.3, while 1 file is linked with Target 10.4. 8 files are linked with Target 10.5, and 99 files are linked with Target 10.7. The frequency of each SDGs Target found in 1,272 legislative files by taking the text of all the articles' as input is shown in Figure 43.

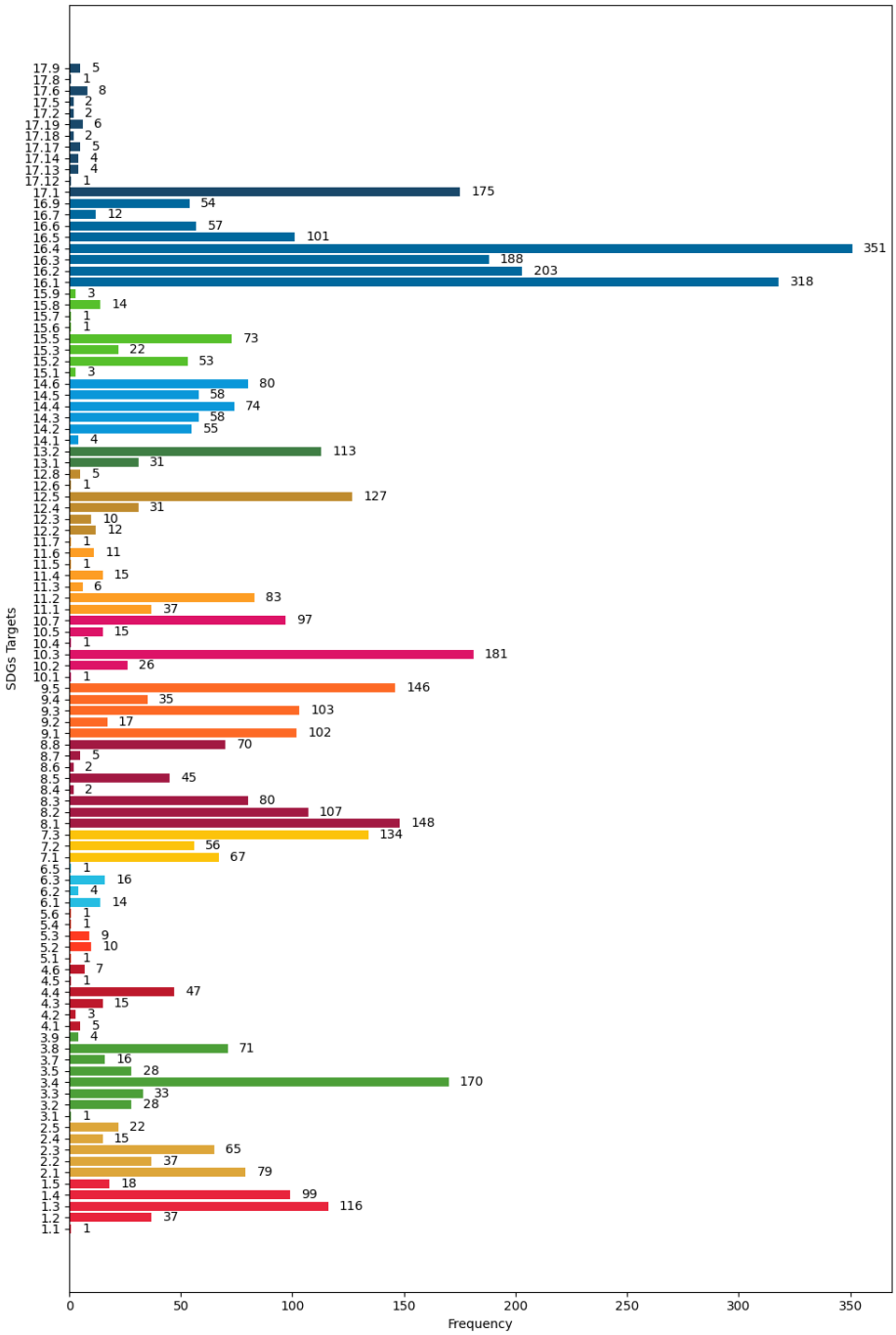

Figure 43: The frequency of each SDGs Target found in 1,272 legislative files by taking the text of all the Articles as input

Target 11.1 is found in 32 files. 66 files are linked with Target 11.2, and 6 are lined with Target 11.3. The 14 files are linked with Target 11.4, and 1 file is linked to Target 11.5. The 8 files are linked with Target 11.6, and 1 file is linked with Target 11.7. 11 files are linked with Target 12.2, while 10 are linked with Target 12.3. 24 files are linked with Target 12.4, and 85 are linked with Target 12.5. 1 file is linked with Target 12.6, and 5 files are linked with Target 12.8.

For Target 13.1, 19 files are linked, while 81 files are linked with Target 13.2. Target 14.1 is linked with 3 files. 26 files are linked with Target 14.2, 52 files are linked with Target 14.3, 69 files are linked with Target 14.4, 27 files are linked with Target 14.5, and 58 files are linked with Target 14.6. The 2 files are linked to Target 15.1, 39 files are linked to Target 15.2, 14 files are linked to Target 15.3, 46 files are linked to Target 15.5 and 2 files are linked to Target 15.6. 1 file is linked to Target 15.7, 13 files are linked to Target 15.8, and 3 files are linked to Target 15.9. For Target 16.1, 293 files are linked, while 183 files are linked with Target 16.2. Target 16.3 is linked with 178 files, and 350 files are linked with Target 16.4. 90 files are linked with Target 16.5, while 48 files are linked with Target 16.6. 10 files are linked with Target 16.7, and 51 files are linked with Target 16.9.

For Target 17.1, 131 files are linked, while 2 files are linked with Target 17.2 and 1 with Target 17.5. 6 files are linked with Target 17.6, and 1 file is linked with Target 17.8. 5 files are linked with Target 17.9, and 1 file is linked with Target 17.12. 2 files are linked with Target 17.13, and 4 files are linked with Target 17.14. Finally, 4 files are linked with Target 17.17, 3 flies are linked with Target 17.18 and 5 files are linked with Target 17.19.

5.2.3 Text of First Four Articles with Preambles

Linking SDGs Targets with legislative files identifies the actions taken by the EU across 1,272 legislative files, with the first four articles having

preambles in the text. This data is fed into the saved SVM model to predict the actions described in the files.

The model linked 1 file with Target 1.1. For Target 1.2, the number of linked files is 37. Target 1.3 is found in 116 files, which are linked with it. The 99 files are linked with Target 1.4, while 18 files are predicted and linked with Target 1.5. Target 2.1 is predicted in 79 files, and all these files are linked with it. Target 2.2 is found in 37 files, which are linked accordingly. The model links 65 files with Target 2.3. For Target 2.4, 15 files are linked. Additionally, 22 legislative files are linked with Target 2.5.

For Target 3.1, 1 file is linked. The 28 files are linked with Target 3.2, while Target 3.3 is linked with 33 files. As per the model's predictions, 170 files are linked with Target 3.4. For Target 3.5, 28 files are linked. Target 3.7 is linked with 16 files, and Target 3.8 is linked with 71 files. Finally, 4 files are linked with Target 3.9.

The 5 files are linked with Target 4.1, while 3 files are linked with Target 4.2. Target 4.3 is predicted in 15 files, and 47 files are linked with Target 4.4. These files are linked based on the model’s predictions, and 1 file is linked with Target 4.5. Target 4.6 is linked with 7 files. For Target 5.1, 1 file is linked, while 10 files are linked with Target 5.2. The 9 files are associated with Target 5.3. The 1 file is linked with Target 5.4, and 1 file is linked with Target 5.6.

The 14 files are linked with Target 6.1, while 4 files are linked with Target 6.2. The 16 files are linked with Target 6.3 based on predictions, and 1 file is linked with Target 6.5. The 67 files are linked with Target 7.1, while 56 files are linked with Target 7.2. For Target 7.3, 134 files are linked based on predictions.

For Target 8.1, 148 files are linked, while 107 files are linked with Target 8.2. The 80 files are linked with Target 8.3, and 2 files are linked with Target 8.4. The 45 files are linked with Target 8.5, and 2 files is linked with Target 8.6. The 5 files are linked with Target 8.7, and 70 files are linked with Target 8.8. The frequency of each SDGs Target found in 1,272 legislative files by taking the text of the four articles' with preambles as input is shown in Figure 44.

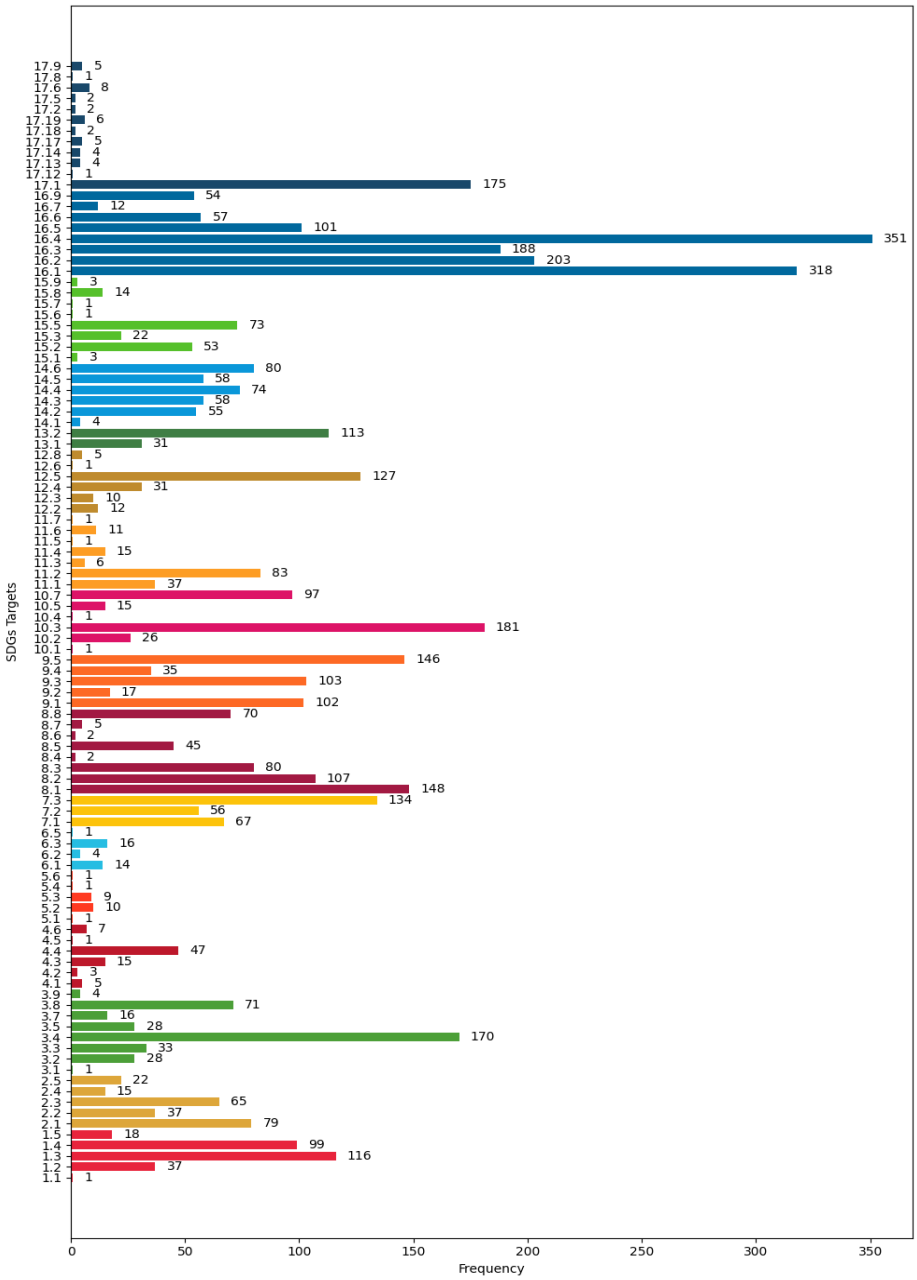

Figure 44: The frequency of each SDG Targets found in 1,272 legislative files by taking text of the first four Articles with Preambles as input

Target 9.1 is found in 102 files, while Target 9.2 is found in 17 files. The 103 files are linked with Target 9.3, and 35 files are linked with Target 9.4. For Target 9.5, 146 files are linked based on predictions. The 1 file is linked with Target 10.1, and 26 files are linked with Target 10.2. A total of 181 files are linked with Target 10.3, while 1 file is linked with Target 10.4. The 15 files are linked with Target 10.5, and 97 files are linked with Target 10.7.

Target 11.1 is found in 37 files. The 83 files are linked with Target 11.2, and 6 files are linked to Target 11.3. The 14 files are linked with Target 11.4, and 1 file is linked to Target 11.5. The 11 files are linked with Target 11.6, and 1 file is linked with Target 11.7. The 12 files are linked with Target 12.2, while 10 are linked with Target 12.3. The 31 files are linked with Target 12.4, and 127 are linked with Target 12.5. The 1 file is linked with Target 12.6, and the 5 files are linked with Target 12.8.

For Target 13.1, 31 files are linked, while 113 files are linked with Target 13.2. Target 14.1 is linked with 4 files. The 55 files are linked with Target 14.2, 58 files are linked with Target 14.3, 74 files are linked with Target 14.4, 58 files are linked with Target 14.5, and 80 files are linked with Target 14.6. The 3 files are linked to Target 15.1, 53 are linked to Target 15.2, 22 are linked to Target 15.3, 73 are linked to Target 15.5, and 1 is linked to Target 15.6. The 1 file is linked to Target 15.7, 14 are linked to Target 15.8, and 3 are linked to Target 15.9.

For Target 16.1, 318 files are linked, while 203 files are linked with Target 16.2. Target 16.3 is linked with 188 files, and 351 files are linked with Target 16.4. The 101 files are linked with Target 16.5, while 57 files are linked with Target 16.6. The 12 files are linked with Target 16.7, and 54 files are linked with Target 16.9.

For Target 17.1, 175 files are linked, while 2 files are linked with Target 17.2 and 2 files with Target 17.5. The 8 files are linked with Target 17.6, and 1 file is linked with Target 17.8. The 5 files are linked with Target 17.9, and 1 file is linked with Target 17.12. The 4 files are linked with Target 17.13, and the 4 files are linked with Target 17.14. Finally,

5 files are linked with Target 17.17, 2 flies are linked with Target 17.18 and 6 files are linked with Target 17.19.

5.2.4 Text of All the Articles with Preambles

Linking legislative files with SDGs Targets that identify the actions taken by the EU across 1,272 legislative files. The text data of all the articles, along with the text of the preamble, is linked with SDGs Targets. The text of all the articles with preambles is fed into the saved SVM model to predict the actions described in the files.

The model linked 1 file with Target 1.1. For Target 1.2, the number of linked files is 47. Target 1.3 is found in 118 files that are linked to it. The 94 files are linked with Target 1.4, while 15 files are predicted and linked with Target 1.5. Target 2.1 is predicted in 76 files, and all these are linked. Target 2.2 is found in 40 files, which are linked accordingly. The model links 67 files with Target 2.3. For Target 2.4, 12 files are linked. Additionally, 19 legislative files are linked with Target 2.5.

For Target 3.1, 1 file is linked. The 26 files are linked with Target 3.2, while Target 3.3 is linked with 23 files. As per the model's predictions, 177 files are linked with Target 3.4. For Target 3.5, 29 files are linked. Target 3.7 is linked with 17 files, and Target 3.8 is linked with 61 files. Finally, 3 files are linked with Target 3.9.

The 5 files are linked with Target 4.1, while 3 files are linked with Target 4.2. Target 4.3 is predicted in 18 files, and 46 files are linked with Target 4.4. These files are linked based on the model’s predictions, and 1 file is linked with Target 4.5. Target 4.6 is linked with 6 files. For Target 5.1, 1 file is linked, while 16 files are linked with Target 5.2. The 8 files are associated with Target 5.3. The 1 file is linked with Target 5.4, and 1 file is linked with Target 5.6. The frequency of each SDGs Target found in 1,272 legislative files by taking the text of all the articles' with preambles text as input is shown in Figure 45.

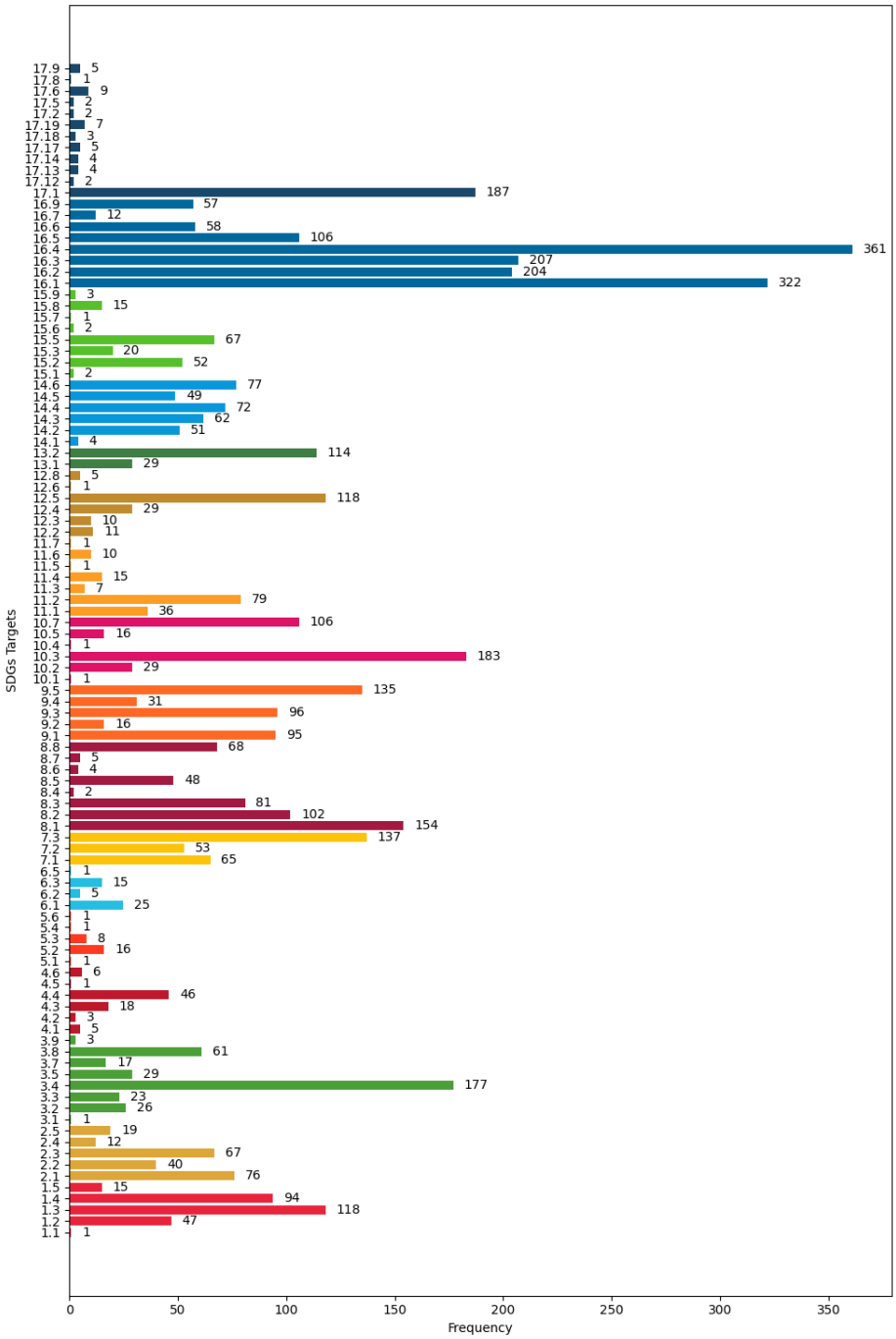

Figure 45: The frequency of each SDGs Target found in 1,272 legislative files by taking the text of all the Articles with Preambles as input

The 25 files are linked with Target 6.1, while 5 files are linked with Target 6.2. The 15 files are linked with Target 6.3 based on predictions, and 1 file is linked with Target 6.5. The 65 files are linked with Target 7.1, while 53 files are linked with Target 7.2. For Target 7.3, 137 files are linked based on predictions.

For Target 8.1, 154 files are linked, while 102 files are linked with Target 8.2. The 81 files are linked with Target 8.3, and 2 files are linked with Target 8.4. The 48 files are linked with Target 8.5, and 4 files is linked with Target 8.6. The 5 files are linked with Target 8.7, and 68 files are linked with Target 8.8.

Target 9.1 is found in 95 files, while Target 9.2 is found in 16 files. The 96 files are linked with Target 9.3, and 31 files are linked with Target 9.4. For Target 9.5, 135 files are linked based on predictions. The 1 file is linked with Target 10.1, and 29 files are linked with Target 10.2. A total of 183 files are linked with Target 10.3, while 1 file is linked with Target 10.4. The 16 files are linked with Target 10.5, and 106 files are linked with Target 10.7.

Target 11.1 is found in 36 files. The 79 files are linked with Target 11.2, and 7 are lined with Target 11.3. The 15 files are linked with Target 11.4 and 1 file is linked with Target 11.5. The 10 files are linked with Target 11.6, and 1 file is linked with Target 11.7. The 11 files are linked with Target 12.2, while 10 are linked with Target 12.3. The 29 files are linked with Target 12.4, and 118 are linked with Target 12.5. The 1 file is linked with Target 12.6, and 5 files are linked with Target 12.8.

For Target 13.1, 29 files are linked, while 114 files are linked with Target 13.2. Target 14.1 is linked with 4 files. The 51 files are linked with Target 14.2, 62 are linked with Target 14.3, 72 are linked with Target 14.4, 49 are linked with Target 14.5, and 77 are linked with Target 14.6.

The 2 files are linked to Target 15.1, 52 files are linked to Target 15.2, 20 files are linked to Target 15.3, 67 files are linked to Target 15.5 and

2 files are linked to Target 15.6. The 1 file is linked to Target 15.7, 15 files are linked to Target 15.8, and 3 files are linked to Target 15.9.

For Target 16.1, 322 files are linked, while 204 files are linked with Target 16.2. Target 16.3 is linked with 207 files, and 361 files are linked with Target 16.4. The 106 files are linked with Target 16.5, while 58 files are linked with Target 16.6. The 12 files are linked with Target 16.7, and 57 files are linked with Target 16.9.

For Target 17.1, 187 files are linked, while 2 files are linked with Target 17.2 and 2 files with Target 17.5. The 9 files are linked with Target 17.6, and 1 file is linked with Target 17.8. The 5 files are linked with Target 17.9, and 2 files are linked with Target 17.12. The 4 files are linked with Target 17.13, and the 4 files are linked with Target 17.14. Finally, 5 files are linked with Target 17.17, 3 flies are linked with Target 17.18 and 7 files are linked with Target 17.19.

5.2.5 All the Text of the File

Linking SDGs Targets with legislative files identifies the actions taken by the EU across 1,272 legislative files. To predict and link the actions at the Target level with the whole text of each file. The whole text data of all the files is fed into the saved SVM model to predict the actions described in the files.

The model linked 1 file with Target 1.1. For Target 1.2, the number of linked files is 42. Target 1.3 is found in 108 files, which are linked with it. The 96 files are linked with Target 1.4, while 16 files are predicted and linked with Target 1.5. Target 2.1 is predicted in 77 files, and all these files are linked with it. Target 2.2 is found in 37 files, which are linked accordingly. The model links 70 files with Target 2.3. For Target 2.4, 11 files are linked. Additionally, 21 legislative files are linked with Target 2.5.

For Target 3.1, 1 file is linked. The 25 files are linked with Target 3.2, while Target 3.3 is linked with 28 files. As per the model's predictions, 179 files are linked with Target 3.4. The frequency of each SDGs Target found in 1,272 legislative files by taking the whole text of each file as input is shown in Figure 46.

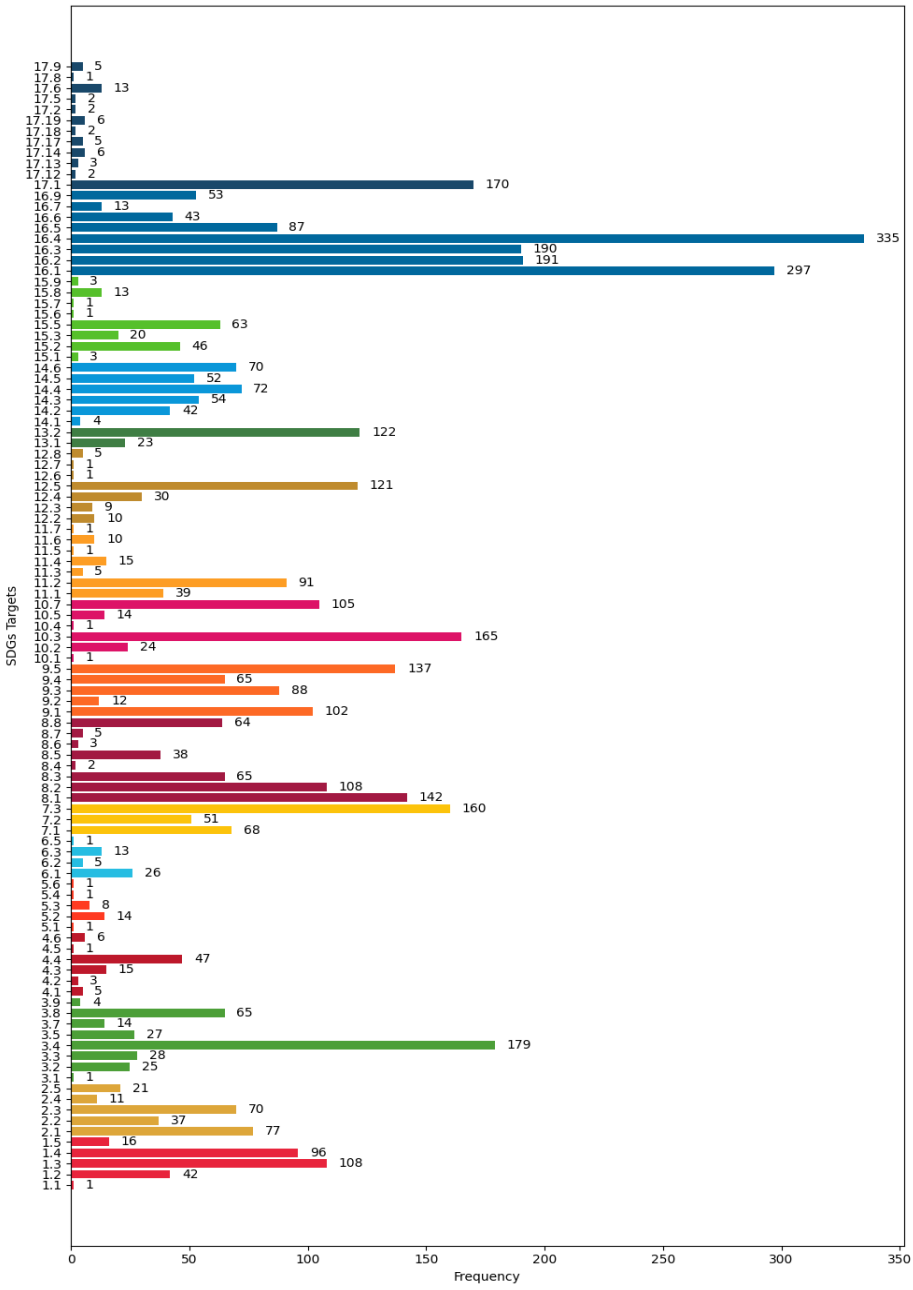

Figure 46: The frequency of each SDGs Target found in 1,272 legislative by taking the whole text of every as input

For Target 3.5, 27 files are linked. Target 3.7 is linked with 14 files, and Target 3.8 is linked with 65 files. Finally, 4 files are linked with Target 3.9. The 5 files are linked with Target 4.1, while 3 files are linked with Target 4.2. Target 4.3 is predicted in 15 files, and 47 files are linked with Target 4.4. These files are linked based on the model's predictions, and 1 file is linked with Target 4.5. Target 4.6 is linked with 6 files. For Target 5.1, 1 file is linked, while 14 files are linked with Target 5.2. The 8 files are associated with Target 5.3. The 1 file is linked with Target 5.4, and 1 file is linked with Target 5.6.

The 26 files are linked with Target 6.1, while 5 files are linked with Target 6.2. The 13 files are linked with Target 6.3 based on predictions, and 1 file is linked with Target 6.5. The 68 files are linked with Target 7.1, while 51 files are linked with Target 7.2. For Target 7.3, 160 files are linked based on predictions. For Target 8.1, 142 files are linked, while 108 files are linked with Target 8.2. 65 files are linked with Target 8.3, and 2 files are linked with Target 8.4. 38 files are linked with Target 8.5, and 3 files are linked with Target 8.6. 5 files are linked with Target 8.7, and 64 files are linked with Target 8.8.

The 3 files are linked to Target 15.1, 46 files are linked to Target 15.2, 20 are linked to Target 15.3, 63 are linked to Target 15.5, and 1 is linked to Target 15.6. The 1 file is linked to Target 15.7. The 13 files are linked to Target 15.8, and 3 files are linked to Target 15.9.

For Target 16.1, 297 files are linked, while 191 files are linked with Target 16.2. Target 16.3 is linked with 190 files, and 335 files are linked with Target 16.4. The 87 files are linked with Target 16.5, while 43 files are linked with Target 16.6. The 13 files are linked with Target 16.7, and 53 files are linked with Target 16.9. For Target 17.1, 170 files are linked, while 2 files are linked with Target 17.2 and 2 files with Target 17.5. The 13 files are linked with Target 17.6, and 1 file is linked with Target 17.8. The 5 files are linked with Target 17.9, and 2 files are linked with Target 17.12. The 3 files are linked with Target 17.13, and the 6 files are linked with Target 17.14. Finally, 5 files are linked with Target 17.17, 2 flies are linked with Target 17.18 and 6 files are linked with Target 17.19.

5.3 Discussion

The analysis of legislative files linked to SDG targets provides valuable insights into the EU's legislative focus on sustainable development. The dataset comprises 1,272 legislative files, with different sections of text (first four articles and with preambles, all articles and with preambles) used to link these files to specific SDG targets.

5.3.1 SDG Goals-Level Linkages

The distribution of linked files varies across SDGs, highlighting the emphasis placed on certain sustainability goals. Notably, SDG 16 (Peace, Justice, and Strong Institutions) has the highest number of linked files, particularly Targets 16.1 (reducing violence), 16.2 (ending abuse and exploitation), 16.3 (promoting rule of law), and 16.4 (combatting illicit financial and arms flows). This suggests that EU legislation is heavily directed toward governance, legal frameworks, and institutional reforms.

Similarly, SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation, and Infrastructure) also have significant legislative coverage, particularly in fostering economic resilience and technological advancements. The presence of a substantial number of files linked to SDG 10 (Reduced Inequalities) and SDG 12 (Responsible Consumption and Production) reflects EU efforts in social equity and sustainability-driven policies.

In contrast, some targets, such as those under SDGs 5 (Gender Equality) and 6 (Clean Water and Sanitation), have relatively fewer linked legislative files. This does not necessarily indicate a lack of attention but could reflect either a lower frequency of direct legislative actions or the integration of these concerns within broader policies rather than standalone legislation.

A comparison between the datasets shows an increase in the number of files linked when using full legislative text. This indicates that a more comprehensive document analysis captures a broader scope of legislative

intent, reinforcing the need for full-text processing in SDG-related research.

Overall, the results highlight the EU's legislative priorities in sustainability, governance, and economic development. However, areas with lower representation might require further policy attention or a different methodological approach to capture indirect legislative links.

5.3.2 SDG Targets-Level Linkages

At the target level, the model provides even more granular insights into the alignment of EU legislative actions with the specific targets of the SDGs. For example, Target 1.3 (aimed at ensuring equal access to resources and opportunities) was linked with 86 definitions, while Target 1.4 (ensuring equal access to land, property, and financial resources) was associated with 77 definitions. This suggests that the EU is actively working to reduce poverty and inequality, but there is also room for more targeted action at a granular level, particularly for areas like property rights and financial inclusion.

Targets associated with SDG Goal 4 (Quality Education), such as Target 4.1 (ensuring that all youth and a substantial portion of adults achieve literacy and numeracy), showed a comparatively lower linkage frequency with just 2 definitions linked to it. This indicates that while education is addressed, further emphasis and integration into EU legislation may be needed.

SDG Goal 13 (Climate Action) and SDG Goal 12 (Responsible Consumption and Production) were linked to Target 13.1 and Target 12.2, respectively, with 2 and 7 definitions. These results indicate that while climate action and responsible consumption are addressed, there is still potential for broader and deeper incorporation into legislative actions.

5.3.3 Areas for Future Development

Despite the impressive linkage between SDG Goals and Targets with legislative files, certain areas could benefit from increased legislative focus. For example, SDG Goal 5 (Gender Equality) and SDG Goal 6

(Clean Water and Sanitation) are relatively underrepresented in the legislative files analyzed. These goals are critical in addressing global inequality and environmental challenges, and increasing their coverage within EU legislation would be beneficial.

Furthermore, while SDG Goal 17 (Partnerships for the Goals) was frequently linked to legislative files, it is essential to deepen the EU's commitment to fostering international partnerships and collaboration. This could be achieved through more detailed legislation and initiatives that focus on global cooperation for sustainable development.

The linkage of SDG Goals and their targets to legislative files provides valuable insights into the alignment of EU law with global sustainable development priorities. The use of machine learning models, particularly SVM, has proven effective in uncovering patterns and relationships in the legislative data. While many SDG goals are well-represented, some areas require greater attention to ensure comprehensive action across all SDGs. Future efforts should focus on reinforcing legislation that supports Gender Equality, Clean Water and Sanitation, and other underrepresented goals, ensuring a more inclusive and sustainable future for all.

Chapter 6
Linking Definitions with Sustainable Development Goals

6 Introduction

The previous chapter presented the process of linking actions identified in EU legislation with the Sustainable Development Goals at the Goal and Target levels. The EU legislative files are linked to SDGs: 17 Goals and 169 Targets. Building on this, the *Definition*s that were annotated using Symbolic AI (see Section 3.10.3) in AKN files were extracted and saved in XLSX file format. Chapter 3 presents a detailed discussion of the annotation of the *Definition*s process.

As discussed in Section 3.10.3, 11,705 *Definition*s were annotated, and those annotated *Delimiting Definition*s were extracted and saved in XLSX file format. After saving these *Definition*s into the XLSX file, this file was provided separately to the best-performing model—at the Goals and Targets levels classification—to link these *Definition*s with the SDG—at the Goals and Target levels—present in these. The remaining details on linking SDGs Goals and Targets level with the *Delimiting Definition*s are discussed in the upcoming sections.

6.1 Linking of Definition with SDGs at Goals Level

Linking *Delimiting Definition*s with SDGs Goals to identify the actions found in EU legislative texts taken by the European Union. The extracted *Delimiting Definition*s are taken as input variables and provided to the best-performing model for classification of SDGs' Goals level. As discussed above (see section 4.4.1.3), the SVM is the best-performing model at the Goal level by incorporating PCA and Class Weights. The *Delimiting Definition*s are given to the saved SVM model to predict the actions present in each *Definition* and link that specific *Definition* with the specific SDGs Goals. All the SDGs Goals found in the *Delimiting Definition*s are predicted, and each *Definition* is linked with that specific Goals based on the predictions.

SDGs Goal 1 (No Poverty) is found in 241 *Definition*s, and these 265 *Definition*s are linked with SDGs Goal 1. SDGs Goal 2 (Zero Hunger) is

predicted in 297 *Definition*s, and based on the prediction, these 297 *Definition*s are linked with Goal 2. SDGs Goal 3 (Good Health and Well-being) is associated with 700 *Definition*s, and these 700 *Definition*s are linked with SDGs Goal 3. With SDGs Goal 4 (Quality Education), 25 *Definition*s are linked based on the model prediction. SDGs Goal 5 (Gender Equality) is associated with 12 *Definition*s, while SDGs Goal 6 (Clean Water and Sanitation) is associated with 10 *Definition*s. Based on these associations, 12 *Definition*s are linked to Goal 5, and 10 *Definition*s are linked to Goal 6. The frequency of each SDGs found in 11705 *Definition*s is shown in Figure 47.

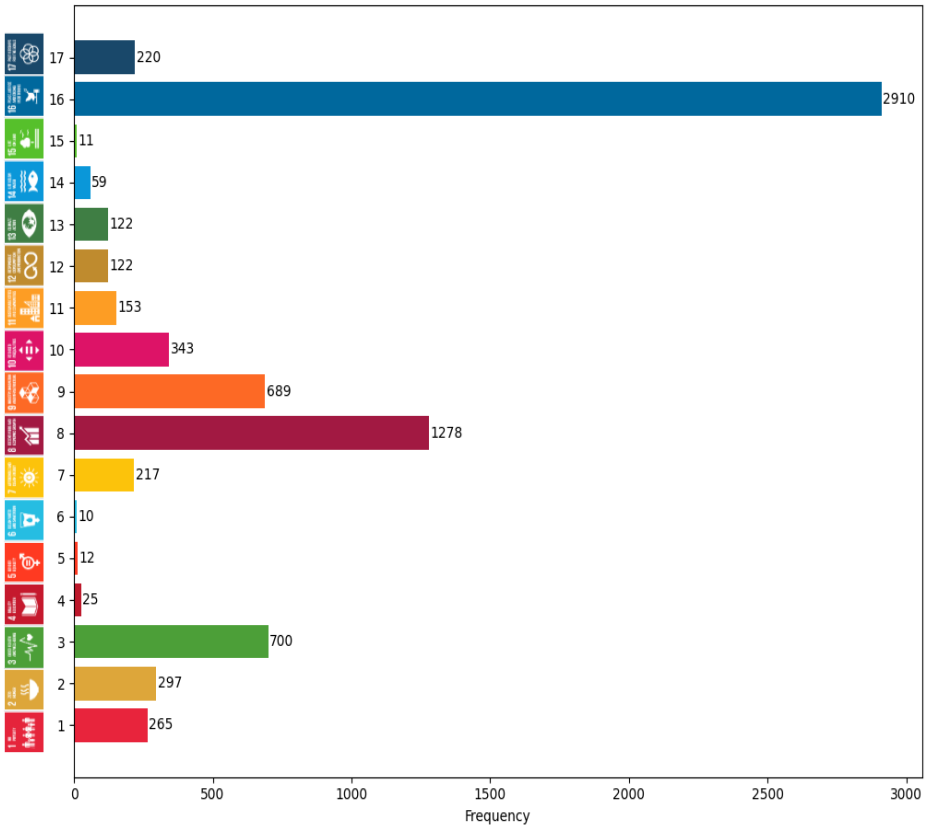

Figure 47: The frequency of each SDGs Goal found in 11705 Delimiting Definition

SDGs Goal 7 (Affordable and Clean Energy) is predicted in 217 *Definition*s, and SDGs Goal 8 (Decent Work and Economic Growth) is connected to 1,278 *Definition*s based on the model prediction. As per the

model's prediction, SDGs Goal 9 (Industry, Innovation, and Infrastructure) is linked with 689 *Definition*s, and SDGs Goal 10 (Reduced Inequality) is linked with 343 *Definition*s. SDGs Goal 11 (Sustainable Cities and Communities) is associated with 153 *Definition*s; based on this, 153 *Definition*s are linked with SDGs Goal 11.

SDGs Goal 12 (Responsible Consumption and Production) is predicted in 122 Definitions, and SDGs Goal 13 (Climate Action) is also found in 122 *Definition*s. As per the model's prediction, SDGs Goal 12 and Goal 13 are linked with 122 *Definition*s.

As predicted by the saved SVM model, 59 definitions are linked to SDGs Goal 14 (Life Below Water), and 11 Definitions are linked to SDGs Goal 15 (Life on Land).

SDGs Goal 16 (Peace, Justice, and Strong Institutions) is associated with 2,910 *Definition*s, and these 2,910 *Definition*s are linked with SDGs Goal 16. SDGs Goal 17 (Partnerships for the Goals) is associated with 220 *Definition*s, and these 220 *Definition*s are linked with SDGs Goal 17.

6.2 Linking of Definition with SDGs at the Target Level

After linking the *Delimiting Definition*s with SDGs at the Goals level, these 11,705 *Delimiting Definition*s are also linked with SDGs Targets. For linking these *Delimiting Definition*s with SDGs Targets, the saved SVM model—selected based on its best performance on the classification of SDGs Targets—is used. The SVM model outperforms others by incorporating PCA for dimensionality reduction and Class Weights to handle class imbalance (see section 4.4.2.4). After saving this model, these 11,705 *Delimiting Definition*s are provided to the saved model to link the actions with them at the Targets level.

The model linked 86 *Definition*s with Target 1.3. For Target 1.4, the number of linked *Definition*s is 77. The model links 5 *Definition*s with Target 1.5. A total of 125 *Definition*s are linked with Target 2.1, while 5

*Definition*s are predicted and linked with Target 2.2. Target 2.3 is predicted in 38 *Definition*s, and all these *Definition*s are linked with it. Target 2.5 is found in 14 *Definition*s, which are linked accordingly.

A total of 43 *Definition*s are linked with Target 3.2, while Target 3.3 is linked with 7 *Definition*s. A total of 175 *Definition*s are linked with Target 3.4 as per the model's predictions. For Target 3.5, 2 *Definition*s are linked. Target 3.7 is linked with 45 *Definition*s, and Target 3.8 is linked with 10 *Definition*s.

Two *Definition*s are linked with Target 4.1, while 1 *Definition* is linked with Target 4.2. For Target 5.2, 16 *Definition*s are linked. Six *Definition* are associated with Target 5.3

Two *Definition*s are linked with Target 6.1, while 9 *Definition*s are linked with Target 6.3. A total of 69 *Definition*s are linked with Target 7.1 based on predictions. A total of 94 *Definition*s are linked with Target 7.2, while 106 *Definition*s are linked with Target 7.3 based on predictions.

For Target 8.1, 239 *Definition*s are linked, while 74 *Definition*s are linked with Target 8.2. A total of 50 *Definition*s are linked with Target 8.3, and 5 *Definition*s are linked with Target 8.4. Four *Definition* are linked with Target 8.5, and 1 *Definition* is linked with Target 8.6. One *Definition* is linked with Target 8.7, and 5 *Definition*s are linked with Target 8.8.

Target 9.1 is found in 41 *Definition*s, and these 41 *Definition*s are linked with Target 9.1. A total of 88 *Definition*s are linked with Target 9.3, and 7 *Definition*s are linked with Target 9.4. For Target 9.5, 45 *Definition*s are linked based on predictions. Six *Definition*s are linked with Target 10.2. A total of 81 *Definition*s are linked with Target 10.3, while 66 *Definition*s are linked with Target 10.5. A total of 24 *Definition*s are linked with Target 10.7. Target 11.1 is found in 21 *Definition*s, and these 21 *Definition*s are connected with Target 11.1. A total of 164 *Definition*s are linked with Target 11.2, and 4 *Definition*s are linked with Target 11.3. Nine *Definition*s are linked with Target 11.4. Another 9 *Definition*s are linked with Target 11.6. The frequency of each SDGs Target found in 11705 *Delimiting Definition*s is shown in Figure 48.

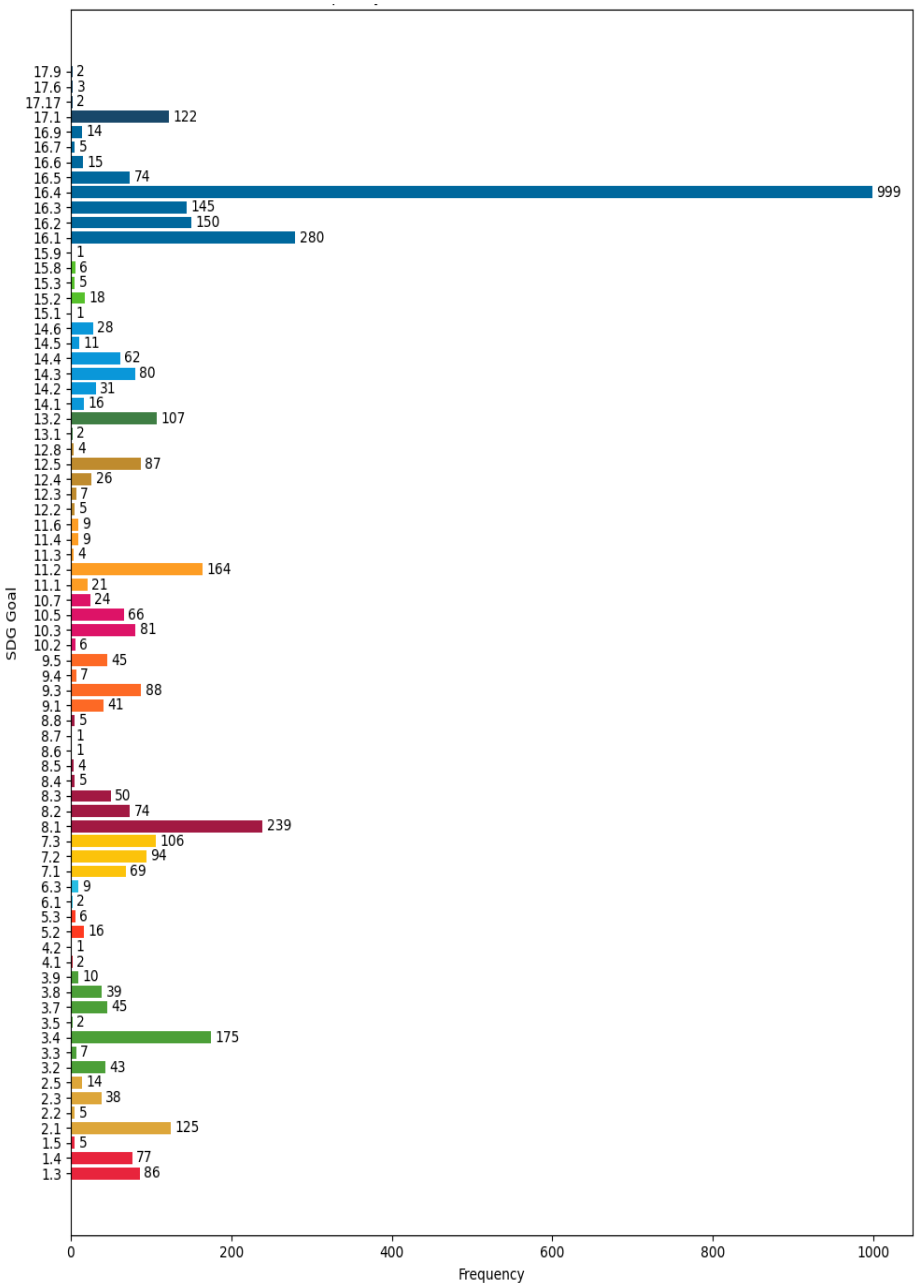

Figure 48: The frequency of each SDGs Targets found in 11705 Delimiting Definition

Five *Definition*s are linked with Target 12.2, while 7 are linked with Target 12.3. A total of 26 *Definition*s are linked with Target 12.4, and 87 are linked with Target 12.5. Four *Definition*s are linked with Target 12.8.

For Target 13.1, 2 *Definition*s are linked, while 107 *Definition*s are linked with Target 13.2. Target 14.1 is linked with 16 *Definition*s. A total of 31 *Definition*s are linked with Target 14.2, 80 *Definition*s are linked with Target 14.3, 62 *Definition*s are linked with Target 14.4, 11 *Definition*s are linked with Target 14.5, and 28 *Definition*s are linked with Target 14.6.

One *Definition* is linked to Target 15.1, 18 *Definition*s are linked to Target 15.2, 5 *Definition*s are linked to Target 15.3, and 6 *Definition*s are linked to Target 15.8. One *Definition* is linked to Target 15.9.

For Target 16.1, 280 *Definition*s are linked, while 150 *Definition*s are linked with Target 16.2. Target 16.3 is linked with 145 *Definition*s, and 999 *Definition*s are linked with Target 16.4. A total of 74 *Definition*s are linked with Target 16.5, while 15 *Definition*s are linked with Target 16.6. Five *Definition*s are linked with Target 16.7, and 14 *Definition*s are linked with Target 16.9.

For Target 17.1, 122 *Definition*s are linked. Three *Definition*s are linked with Target 17.6. Two *Definition*s are linked with Target 17.9. Finally, two *Definition*s are linked with Target 17.17.

6.3 Discussion

The linking of Delimiting Definitions in EU legislation to Sustainable Development Goals (SDGs) at both the Goal and Target levels provides a deeper understanding of how legal definitions contribute to sustainable policy implementation. Through the use of Symbolic AI and supervised classification with Support Vector Machines (SVM), 11,705 annotated Definitions were systematically categorized to reveal their alignment with SDGs.

At the Goal level, the analysis highlights that SDG 16 (Peace, Justice, and Strong Institutions) has the highest number of linked Definitions (2,910), reinforcing the EU's legislative emphasis on governance, legal

integrity, and institutional stability. Similarly, SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation, and Infrastructure) show strong representation with 1,278 and 689 linked Definitions, respectively, reflecting a focus on economic resilience and industrial development. In contrast, SDGs related to environmental sustainability (e.g., SDG 6: Clean Water and Sanitation, SDG 14: Life Below Water, and SDG 15: Life on Land) exhibit significantly lower numbers of linked Definitions, suggesting either a lesser volume of legal Definitions addressing these areas or a broader integration of sustainability concerns into non-definition-based legal frameworks.

At the Target level, the classification model provides more granular insights into how specific legal Definitions map onto SDG Targets. Notably, Target 16.4 (combatting illicit financial and arms flows) has the highest number of linked Definitions (999), again emphasizing the EU's legislative commitment to institutional transparency and legal enforcement. Other highly linked Targets include 8.1 (sustained economic growth), 7.3 (energy efficiency), and 13.2 (climate policy integration), reflecting the EU's broader economic and environmental priorities. However, several Targets (e.g., 4.1: Quality Education, 5.2: Eliminating Violence Against Women, and 6.1: Universal Access to Safe Drinking Water) have significantly fewer linked Definitions, potentially indicating gaps in explicit legal definitions supporting these policy areas.

The results demonstrate the effectiveness of machine learning in structuring and analyzing legal texts for SDG alignment. The SVM model, enhanced with Principal Component Analysis (PCA) and class weighting, successfully identified patterns in legal Definitions that correspond to sustainable development objectives. However, the observed disparities in linkage frequency across different Goals and Targets suggest potential areas for further legal scrutiny and policy enhancement.

Overall, this study underscores the importance of AI-driven methodologies in legislative analysis, offering policymakers data-driven insights to refine legal frameworks for achieving SDG commitments.

Future work could explore extending this approach to additional legal texts or integrating neural-symbolic AI to enhance interpretability in the classification process.

Chapter 7
Conclusion and Future Work

7 Conclusion

This research focuses on developing ‘Legimatics and AI Tools for the Monitoring of EU Legislation in Agrifood and Sustainable Development Goals’. The first objective is annotating legal norms in EU legislative documents using Symbolic AI. Specifically, the study targeted the annotation of *Delimiting Definition*s, with each annotation comprising a "definition heading" and a "definition body." The dataset included 15,082 EU legislative files in AKN.

The first task is annotation, which is divided into two scenarios. In the first scenario, *Definition*s within articles under a specific "heading" tag containing the inner text ‘Definition or Definitions’ are annotated. In the second, independent *Delimiting Definition*s are annotated wherever they appear in the legislative files. Annotation algorithms, implemented in Python using the ElementTree library, employed rule-based mining to identify Targeted text. Once detected, each definition's heading, definition body, and subparts were annotated with tags such as def and defBody. The annotated data is then validated through indentation checks in the AKN format. Using the first algorithm, 899 files are successfully annotated, while the second scenario annotated 1,272 files. The remaining files did not fulfil the basic requirements of annotations and did not find any annotated material, so those were unchanged.

In total, 11,705 *Definition*s are annotated, with 92.88% found within articles under the heading tag with the inner text ‘Definition or Definition’. Following the annotation process, Named Entity Recognition (NER) is applied to identify and label locations, dates, and times.

After completing the annotation, the second task is to develop a multilabel classification model to link EU legislative texts to SDG Goals and Targets. This involved implementing Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) classifiers for multilabel classification. Various preprocessing techniques were evaluated, including lowercasing, removing stop words, eliminating special characters & punctuation, stemming, lemmatization, and removing short words. The combination of lowercasing and removing stop words yielded the best results.

TF/IDF, alongside Unigram, Bigram, and Trigram methods, are tested for feature extraction. TF/IDF performed the best, so TF/IDF is incorporated. Dimensionality reduction techniques such as PCA, t-SNE, and UMAP are explored, with PCA providing optimal results. Consequently, preprocessing included lowercasing and removing stop words, feature extraction using TF/IDF, and dimensionality reduction applied PCA for both SVM and KNN classifiers. Class Weights are integrated into the SVM model to address the class imbalance, as KNN does not support this feature.

To develop a classification model, the dataset was scraped from KnowSDGs initiatives and preprocessed by separating Directives (L) and Regulations (R) for ML model development. The final dataset is split into a 2:1 ratio using the Stratified sampling method to train and test the model. SVM is tested in four configurations. For SDGs classification at the Goal level, the combination of PCA and Class Weights applied to the first four articles with preambles achieved the best performance, with an accuracy of 63.44% and an F-score of 67.50%. At the Target level, SVM with PCA and Class Weights applied to all articles with preambles performed better, achieving 52.04% accuracy and an F-score of 56.22%. The best-performing models were saved to facilitate the linking of EU legislation with SDGs at both the Goal and Target levels.

The third task involves linking EU legislative actions to SDGs Goals and Targets in two scenarios. In the first, legislative text is used for linking, while in the second, annotated Delimiting Definitions are employed. The first scenario linking is performed using five setups: (1) the first four articles' text, (2) all articles' text, (3) the first four articles with preambles text, (4) all articles with preambles text, and (5) the entire file text. In the second scenario, annotated Delimiting Definitions are linked to SDG Goals and Targets.

By integrating annotation, classification, and linking of EU legislation with Sustainable Development Goals at the Goals and Targets Level, this research provides a robust mechanism for policymakers and researchers

to monitor legislative alignment with Sustainable Development Goals objectives, enabling informed decision-making and effective policy formulation.

7.1 Match the RQs with the Solutions

As described in the introduction of this Thesis (see Section 1.4), the collection of works described in the previous Chapters addresses the following main research questions:

- RQ1: How can norms be detected and annotated from EU legislative text files?
- RQ2: How can a classification model be developed to classify EU legislative text into the Sustainable Development Goals?
- RQ3: How can the efficacy of rules and norms be monitored to facilitate decision-making and policy formulation while identifying actions linked to the SDGs?

7.1.1 RQ1 Results and the Solution

To address RQ1, this study developed a comprehensive framework for detecting and annotating legal norms from EU legislative text files using Symbolic AI. The annotation process Targeted *Delimiting Definition*s, incorporating a "definition heading" and a "definition body" within legislative texts. The dataset, consisting of 15,082 legislative files, was preprocessed into the AKN file format to standardize the structure. Two annotation scenarios were implemented: (1) annotating definitions within articles under specific "heading" tag containing inner text ‘Definition/Definitions’ (see section 3.9.1), and (2) annotating independent *Delimiting Definition*s wherever they appeared (see section 3.9.2). Rule-based mining techniques, supported by algorithms implemented in Python's ElementTree library, were employed to identify and tag Targeted *Definition*s with def and defBody tags. The validation of annotated files, verified through manual checks (see section 3.11.1) and inter-annotator agreement (Cohen’s Kappa score of 0.972), confirmed the accuracy and reliability of the process (see section 3.11.2.4). A total of 11,705

*Definition*s were annotated, providing a robust foundation for further analysis and monitoring of legal norms.

7.1.2 RQ2 Results and the Solution

The solution to RQ2 involved the development of a multilabel classification model to link EU legislative text to SDG Goals and Targets. Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) classifiers were implemented for this purpose. Preprocessing techniques such as lowercasing, stop-word removal, and punctuation elimination were evaluated, with lowercasing and stop-word removal yielding the best results (see section 4.2.1.2.1). Feature extraction was conducted using TF/IDF alongside Unigram, Bigram, and Trigram methods, with TF/IDF demonstrating superior performance (see section 4.3.2.3). Dimensionality reduction techniques, including PCA, t-SNE, and UMAP, were tested, with PCA emerging as the most effective (see section 4.3.2.4).

Class Weights were integrated into the SVM model to address the class imbalance (see section 4.3.2.5). The dataset scraped from KnowSDG's (see section 4.2.1.2), was preprocessed and divided into training and testing subsets in a 2:1 ratio (see section 4.2.1.2.2), with the testing dataset oversampled to balance the classes (see section 4.2.1.2.3). At the Goals level, the best performance was achieved by applying SVM with PCA and Class Weights to the first four articles with preambles, resulting in an accuracy of 53.34%, a weighted F-score of 70.04% and with macro F-score of 57.94% (see section 4.4.1.3). At the Target level, the same combination applied to all articles with preambles yielded an accuracy of 46.56% accuracy, a weighted F-score of 56.0% and 30.61% macro F-score (see section 4.4.2.4). These models were saved to establish a robust link between EU legislation and SDGs (see section 4.5).

7.1.3 RQ3 Results and the Solution

To address RQ3, the study linked the annotated *Definition*s (see Chapter 6) and the legislative files (see Chapter 5) to SDGs in two scenarios. In the first scenario, the linkage was achieved using the text of legislative files across five configurations: (1) the first four articles' text (see section 5.1.1 and section 5.2.1), (2) all articles' text (see section 5.1.2 and section 5.2.2), (3) the first four articles with preambles text (see section 5.1.3 and section 5.2.3), (4) all articles with preambles text (see section 5.1.4 and section 5.2.4), and (5) the entire text of each legislative file (see section 5.1.5 and section 5.2.5). In the second scenario, the annotated *Delimiting Definition*s were used for linking with SDGs at the Goal level (see section 6.1) and Target Level (see section 6.2).

The models developed in response to RQ2 facilitated this linkage by enabling the classification of legislative texts and definitions at both the Goal and Target levels. By automating the association of legislative actions with SDGs objectives, this framework provides a mechanism to monitor the efficacy of rules and norms. This supports informed decision-making and policy formulation, ensuring alignment with the objectives of Sustainable Development Goals.

7.2 Future Work

This research can further be extended by developing an automated annotation tool or technique capable of efficiently annotating legislative files and legal norms with SDGs (at Goals, Targets and Indicators level). Such a tool would facilitate processing large datasets, ensuring balanced class distributions, which is essential for optimal model training and testing. This would enable the development of models that more accurately link the actions taken by the European Union to the United Nations' Sustainable Development Goals agenda.

Another promising avenue is creating a real-time monitoring system to continuously analyze newly introduced legislative texts and update their alignment with SDGs. This system could integrate temporal

analysis to track the evolution of legislative contributions to the SDGs over time, providing valuable insights into the impact and progress of policies.

By addressing these directions, this research has the potential to evolve into a comprehensive and adaptable framework for monitoring and aligning legislative actions with SDGs objectives. Ultimately, it could play a crucial role in supporting evidence-based policymaking and advancing sustainable development initiatives at both regional and global levels.

8 References

[1] A. Żak, “Multiple perspectives on sustainable development,” in *Organizing Sustainable Development*, Taylor and Francis, 2023, pp. 77–90. doi: 10.4324/9781003379409-8.

[2] N. Chaaben, Z. Elleuch, B. Hamdi, and B. Kahouli, “Green economy performance and sustainable development achievement: empirical evidence from Saudi Arabia,” *Environ Dev Sustain*, vol. 26, no. 1, pp. 549–564, Jan. 2024, doi: 10.1007/s10668-022-02722-8.

[3] I. M. Ziaul and W. Shuwei, “ENVIRONMENTAL SUSTAINABILITY: A MAJOR COMPONENT OF SUSTAINABLE DEVELOPMENT Volume: 4 Number: 3 Page: 900-907.”

[4] A. Ziai, “Beyond the Sustainable Development Goals: Post-development Alternatives,” 2024, pp. 35–54. doi: 10.1007/978-3-031-30308-1_3.

[5] O. Adeboye, “7 ‘A starving man cannot shout halleluyah’ African Pentecostal Churches and the challenge of promoting sustainable development.”

[6] R. Russell-Bennett, M. S. Rosenbaum, R. P. Fisk, and M. M. Raciti, “SDG editorial: improving life on planet earth – a call to action for service research to achieve the sustainable development goals (SDGs),” *Journal of Services Marketing*, vol. 38, no. 2, pp. 145–152, Jan. 2024, doi: 10.1108/JSM-11-2023-0425.

[7] B. Liang, J. Cao, J. He, S. Li, X. Zhang, and J. Ou, “Prioritization Decision-making Model for Sustainable Development Goals Based on Multi-factor Simulated Annealing Particle Swarm Algorithm,” *SCIREA Journal of Computer*, Oct. 2019, doi: 10.54647/computer520400.

[8] K. Berbeka, W. Alejziak, and J. Berbeka, “Sustainable development goals of Agenda 2030 in the declarations and aims of international tourism organisations,” *Journal of Travel and Tourism Marketing*, vol. 41, no. 1, pp. 142–153, 2024, doi: 10.1080/10548408.2023.2239862.

[9] F. Piadeh *et al.*, “A critical review for the impact of anaerobic digestion on the sustainable development goals,” Jan. 01, 2024, *Academic Press*. doi: 10.1016/j.jenvman.2023.119458.

[10] P. Heriyati, N. Yadav, and D. Tamara, “Accomplishing sustainable development goals through international management system standards and multinational supply chains,” *Bus Strategy Environ*, 2024, doi: 10.1002/bse.3752.

[11] S. M. Ahmed, A. A. Elbushra, A. E. Ahmed, A. H. El-Meski, and K. O. Awad, “Climate variability impacts on crop yields and agriculture contributions to gross domestic products in the Nile basin (1961–2016): What did deep machine learning algorithms tell us?,” *Theor Appl Climatol*, 2024, doi: 10.1007/s00704-024-04858-1.

[12] S. Goubran, T. Walker, C. Cucuzzella, and T. Schwartz, “Green building standards and the United Nations’ Sustainable Development Goals,” *J Environ Manage*, vol. 326, Jan. 2023, doi: 10.1016/j.jenvman.2022.116552.

[13] M. Alenezi and M. Akour, “Digital Transformation Blueprint in Higher Education: A Case Study of PSU,” *Sustainability (Switzerland)*, vol. 15, no. 10, May 2023, doi: 10.3390/su15108204.
[14] S. Plagerson, *Human well-being and capabilities*.
[15] T. Cernev and R. Fenner, “The importance of achieving foundational Sustainable Development Goals in reducing global risk,” *Futures*, vol. 115, Jan. 2020, doi: 10.1016/j.futures.2019.102492.
[16] M. Mumtaz, “Role of civil society organizations for promoting green and blue infrastructure to adapting climate change: Evidence from Islamabad city, Pakistan,” *J Clean Prod*, vol. 309, Aug. 2021, doi: 10.1016/j.jclepro.2021.127296.
[17] Dr. S. K. Pal, “THE IMPACT OF INCREASE IN COVID-19 CASES WITH EXCEPTIONAL SITUATION TO SDG: GOOD HEALTH AND WELL-BEING,” *Ann Trop Med Public Health*, vol. 23, no. 17, 2020, doi: 10.36295/asro.2020.232018.
[18] H. Valin, T. Hertel, B. L. Bodirsky, T. Hasegawa, and E. Stehfest, “A paper from the Scientific Group of the UN Food Systems Summit Achieving Zero Hunger by 2030 A Review of Quantitative Assessments of Synergies and Tradeoffs amongst the UN Sustainable Development Goals,” 2021. [Online]. Available: https://sc-fss2021.org/
[19] P. Atukunda, W. B. Eide, K. R. Kardel, P. O. Iversen, and A. C. Westerberg, “Unlocking the potential for achievement of the un sustainable development goal 2 – ‘zero hunger’ – in Africa: Targets, strategies, synergies and challenges,” 2021, *Swedish Nutrition Foundation*. doi: 10.29219/fnr.v65.7686.
[20] J. Woodhill, A. Kishore, J. Njuki, K. Jones, and S. Hasnain, “Food systems and rural wellbeing: challenges and opportunities,” *Food Secur*, vol. 14, no. 5, pp. 1099–1121, Oct. 2022, doi: 10.1007/s12571-021-01217-0.
[21] T. Sjah and Z. Zainuri, “Agricultural Supply Chain and Food Security,” 2020, pp. 79–88. doi: 10.1007/978-3-319-95675-6_82.
[22] “enhancing_livestocks_contribution_to_one_health_a-wageningen_university_and_research_646723”.
[23] A. Gatto, “Quantifying management efficiency of energy recovery from waste for the circular economy transition in Europe,” *J Clean Prod*, vol. 414, Aug. 2023, doi: 10.1016/j.jclepro.2023.136948.
[24] P. Nakhle, I. Stamos, P. Proietti, and A. Siragusa, “Environmental monitoring in European regions using the sustainable development goals (SDG) framework,” *Environmental and Sustainability Indicators*, vol. 21, Feb. 2024, doi: 10.1016/j.indic.2023.100332.
[25] S. O. Park and N. Hassairi, *What predicts legislative success of early care and education policies?: Applications of machine learning and natural language processing in a cross-state early childhood policy analysis*, vol. 16, no. 2 Febuary. 2021. doi: 10.1371/journal.pone.0246730.

[26] J. Chen, B. Goudey, J. Zobel, N. Geard, and K. Verspoor, “Exploring automatic inconsistency detection for literature-based gene ontology annotation,” *Bioinformatics*, vol. 38, pp. I273–I281, 2022, doi: 10.1093/bioinformatics/btac230.

[27] A. Pegan, “A temporal perspective on staff support in the European parliament,” *J Eur Integr*, vol. 44, no. 4, pp. 511–529, 2022, doi: 10.1080/07036337.2021.1942463.

[28] A. Lähteenmäki-Uutela, M. Rahikainen, M. T. Camarena-Gómez, J. Piiparinen, K. Spilling, and B. Yang, “European Union legislation on macroalgae products,” *Aquaculture International*, vol. 29, no. 2, pp. 487–509, 2021, doi: 10.1007/s10499-020-00633-x.

[29] M. Cherubini, F. Romano, A. Bolioli, L. De Mattei, and M. Sangermano, “Improving the accessibility of EU laws: the Chat-EUR-Lex project,” *CEUR Workshop Proc*, vol. 3762, pp. 6–11, 2024.

[30] M. Ovádek, “Facilitating access to data on European Union laws,” *Political Research Exchange*, vol. 3, no. 1, 2021, doi: 10.1080/2474736X.2020.1870150.

[31] M. Avgerinos Loutsaris, C. Alexopoulos, M. I. Maratsi, and Y. Charalabidis, “Semantic Interoperability for Legal Information: Mapping the European Legislation Identifier (ELI) and Akoma Ntoso (AKN) Ontologies,” *ACM International Conference Proceeding Series*, pp. 41–53, 2023, doi: 10.1145/3614321.3614327.

[32] M. Palmirani and D. Liga, “Derogations Analysis of European Legislation Through Hybrid AI Approach,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13429 LNCS, pp. 123–137, 2022, doi: 10.1007/978-3-031-12673-4_9.

[33] E. Filtz, S. Kirrane, and A. Polleres, “The linked legal data landscape: linking legal data across different countries,” *Artif Intell Law (Dordr)*, vol. 29, no. 4, pp. 485–539, 2021, doi: 10.1007/s10506-021-09282-8.

[34] M. Avgerinos Loutsaris, Z. Lachana, C. Alexopoulos, and Y. Charalabidis, “Legal Text Processing: Combing two legal ontological approaches through text mining,” *ACM International Conference Proceeding Series*, pp. 522–532, 2021, doi: 10.1145/3463677.3463730.

[35] M. Haag, S. Hurka, and C. Kaplaner, “Policy complexity and implementation performance in the European Union,” *Regul Gov*, no. January, 2024, doi: 10.1111/rego.12580.

[36] R. Ferrod, D. A. Bondarenko, D. Audrito, and G. Siragusa, “Pairing EU directives and their national implementing measures: A dataset for semantic search,” *Computer Law and Security Review*, vol. 51, p. 105862, 2023, doi: 10.1016/j.clsr.2023.105862.

[37] T. König, B. Luetgert, and T. Dannwolf, “Quantifying European legislative research: Using CELEX and PreLex in EU legislative studies,” *Eur Union Polit*, vol. 7, no. 4, pp. 553–574, 2006, doi: 10.1177/1465116506069444.

[38] A. Stellato and M. Fiorelli, “LegalHTML: A Representation Language for Legal Acts,” *Lecture Notes in Computer Science (including subseries Lecture*

Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 13870 LNCS, pp. 520–537, 2023, doi: 10.1007/978-3-031-33455-9_31.

[39] M. Palmirani *et al.*, "Legal Drafting supported by AI: enhancing LEOS," *CEUR Workshop Proc*, vol. 3762, pp. 458–463, 2024.

[40] M. Palmirani, F. Sovrano, D. Liga, S. Sapienza, and F. Vitali, "Hybrid AI Framework for Legal Analysis of the EU Legislation Corrigenda," *Frontiers in Artificial Intelligence and Applications*, vol. 346, pp. 68–75, 2021, doi: 10.3233/FAIA210319.

[41] M. Asif and M. Palmirani, "Legal Definition Annotation in EU Legislation Using Symbolic AI," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2024, pp. 34–39. doi: 10.1007/978-3-031-68211-7_4.

[42] I. Munyoro, "Assessing Parliament of Zimbabwe's informatics database as a tool for providing evidence-based information for decision making," *Journal of Librarianship and Information Science*, vol. 51, no. 1, pp. 218–227, 2019, doi: 10.1177/0961000617726122.

[43] F. Fitsilis, D. Koryzis, and G. Schefbeck, "Legal Informatics Tools for Evidence-Based Policy Creation in Parliaments," *International Journal of Parliamentary Studies*, vol. 2, no. 1, pp. 5–29, 2022, doi: 10.1163/26668912-bja10031.

[44] S. Leventis, F. Fitsilis, and V. Anastasiou, "Diversification of legislation editing open software (Leos) using software agents—transforming parliamentary control of the hellenic parliament into big open legal data," *Big Data and Cognitive Computing*, vol. 5, no. 3, 2021, doi: 10.3390/bdcc5030045.

[45] G. Wayne, "Effective Drafting for Effective Legislation: Utilising Thornton's Five Stages of Drafting in Papua New Guinea," Institute of Advanced Legal Studies, School of Advanced Study, University of~…, 2016.

[46] A. Cvejić, K. Grujić, A. Cvejić, M. Marković, and S. Gostojić, "Automatic Transformation of Plain-text Legislation into Machine-readable Format," *Proceedings of the 11th International Conference on Information Society and Technology*, no. October, pp. 50–55, 2021, [Online]. Available: https://www.eventiotic.com/eventiotic/library/paper/638

[47] M. A. Qureshi *et al.*, "A Novel Auto-Annotation Technique for Aspect Level Sentiment Analysis," *CMC-COMPUTERS MATERIALS \& CONTINUA*, vol. 70, no. 3, pp. 4987–5004, 2022.

[48] M. A. Qureshi *et al.*, "Aspect Level Songs Rating Based Upon Reviews in English," *Computers, Materials and Continua*, vol. 74, no. 2, pp. 2589–2605, 2023, doi: 10.32604/cmc.2023.032173.

[49] S. Zheng *et al.*, "Joint entity and relation extraction based on a hybrid neural network," *Neurocomputing*, vol. 257, pp. 59–66, 2017.

[50] K. Adnan and R. Akbar, “An analytical study of information extraction from unstructured and multidimensional big data,” *J Big Data*, vol. 6, no. 1, pp. 1–38, 2019.

[51] B. Oral and G. Eryi\ugit, “Fusion of visual representations for multimodal information extraction from unstructured transactional documents,” *International Journal on Document Analysis and Recognition (IJDAR)*, pp. 1–19, 2022.

[52] M. Faxriddin Qizi Samiyeva and M. Abdulla Qizi Madyarova, “Text mining and it is development stages.” [Online]. Available: www.openscience.uz

[53] E. Senave, M. J. Jans, and R. P. Srivastava, “The application of text mining in accounting,” *International Journal of Accounting Information Systems*, vol. 50, p. 100624, Sep. 2023, doi: 10.1016/j.accinf.2023.100624.

[54] N. Farzana, P. Mohammed1, F. R. Shaikh2, T. Priya3, K. Khan4, and S. Jamadar5, “Obtaining and Analyzing Data from Texts,” vol. 12, p. 2023.

[55] I. Ihsan, H. Rahman, A. Shaikh, A. Sulaiman, K. Rajab, and A. Rajab, “Improving in-text citation reason extraction and classification using supervised machine learning techniques,” *Comput Speech Lang*, vol. 82, Jul. 2023, doi: 10.1016/j.csl.2023.101526.

[56] T. Gupta, M. Zaki, N. M. A. Krishnan, and Mausam, “MatSciBERT: A materials domain language model for text mining and information extraction,” *NPJ Comput Mater*, vol. 8, no. 1, Dec. 2022, doi: 10.1038/s41524-022-00784-w.

[57] C. Baden, C. Pipal, M. Schoonvelde, and M. A. C. G. van der Velden, “Three Gaps in Computational Text Analysis Methods for Social Sciences: A Research Agenda,” *Commun Methods Meas*, vol. 16, no. 1, pp. 1–18, 2022, doi: 10.1080/19312458.2021.2015574.

[58] A. Kumar, V. Dabas, and P. Hooda, “Text classification algorithms for mining unstructured data: a SWOT analysis,” *International Journal of Information Technology*, vol. 12, no. 4, pp. 1159–1169, 2020.

[59] S. Singh, “Natural language processing for information extraction,” *arXiv preprint arXiv:1807.02383*, 2018.

[60] D. Toluhi, R. Schmidt, and B. Parsia, “Concept Description and Definition Extraction for the ANEMONE System,” in *Engineering Multi-Agent Systems: 9th International Workshop, EMAS 2021, Virtual Event, May 3--4, 2021, Revised Selected Papers*, 2022, pp. 352–372.

[61] J. Cindrič, L. Kuhelj, S. Sever, Ž. Simonišek, and M. Šemen, “Data Collection and Definition Annotation for Semantic Relation Extraction”.

[62] S. Bokota, “Defining human-animal chimeras and hybrids: A comparison of legal systems and natural sciences,” 2021.

[63] R. M. Haig, “The concept of income—economic and legal aspects,” in *Forerunners of Realizable Values Accounting in Financial Reporting*, Routledge, 2020, pp. 140–167.

[64] D. Kuhn, “Critical thinking as discourse,” *Hum Dev*, vol. 62, no. 3, pp. 146–164, 2019.

[65] B. S. Case, “Riots as Civil Resistance: Rethinking the Dynamics of ‘Nonviolent’Struggle,” *Journal of Resistance Studies*, vol. 4, no. 1, pp. 9–44, 2018.

[66] G. Brochmann and T. Hammar, *Mechanisms of immigration control: A comparative analysis of European regulation policies*. Routledge, 2020.
[67] N. Gardner, H. Khan, and C.-C. Hung, "Definition modeling: literature review and dataset analysis," *Applied Computing and Intelligence*, vol. 2, no. 1, pp. 83–98, 2022, doi: 10.3934/aci.2022005.
[68] A. Zaki-Ismail, M. Osama, M. Abdelrazek, J. Grundy, and A. Ibrahim, "RCM-extractor: an automated NLP-based approach for extracting a semi formal representation model from natural language requirements," *Automated Software Engineering*, vol. 29, no. 1, pp. 1–33, 2022.
[69] F. Hamborg, "Towards Automated Frame Analysis: Natural Language Processing Techniques to Reveal Media Bias in News Articles," 2022.
[70] A. P. Ben Veyseh, F. Dernoncourt, D. Dou, and T. H. Nguyen, "A joint model for definition extraction with syntactic connection and semantic consistency," *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pp. 9098–9105, 2020, doi: 10.1609/aaai.v34i05.6444.
[71] L. Claassen *et al.*, "Cold brew coffee—Pilot studies on definition, extraction, consumer preference, chemical characterization and microbiological hazards," *Foods*, vol. 10, no. 4, p. 865, 2021.
[72] P. Kumar, A. Singh, P. Kumar, and C. Kumar, "An explainable machine learning approach for definition extraction," in *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences*, 2020, pp. 145–155.
[73] S. Spala, N. A. Miller, F. Dernoncourt, and C. Dockhorn, "SemEval-2020 Task 6: Definition extraction from free text with the DEFT corpus," *14th International Workshops on Semantic Evaluation, SemEval 2020 - co-located 28th International Conference on Computational Linguistics, COLING 2020, Proceedings*, no. 2006, pp. 336–345, 2020, doi: 10.18653/v1/2020.semeval-1.41.
[74] A. Kanapala, S. Pal, and R. Pamula, "Text summarization from legal documents: a survey," *Artif Intell Rev*, vol. 51, no. 3, pp. 371–402, 2019, doi: 10.1007/s10462-017-9566-2.
[75] C. Niculi\ct\ua and L. Dumitriu, "The Relational Parts of Speech in Text Analysis for Definition Detection, for Romanian Language," in *2019 18th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, 2019, pp. 1–6.
[76] H. Kelsen and A. J. Trevino, *General theory of law \& state*. Routledge, 2017.
[77] M. Allahyari *et al.*, "A brief survey of text mining: Classification, clustering and extraction techniques," *arXiv preprint arXiv:1707.02919*, 2017.
[78] E. A. Varó and B. Hughes, *Legal translation explained*. Routledge, 2014.
[79] P. Olsen and M. Borit, "How to define traceability," *Trends Food Sci Technol*, vol. 29, no. 2, pp. 142–150, 2013.
[80] F. A. Hayek, *Law, legislation and liberty, volume 1: Rules and order*. University of Chicago Press, 2011.

[81] I. Iosim, M. Seracin, G. Popescu, and others, “Definiendum and definientia of" communication".,” *Lucr{\u{a}}ri Ştiinţifice, Universitatea de Ştiinţe Agricole Şi Medicin{\u{a}} Veterinar{\u{a}} a Banatului, Timisoara, Seria I, Management Agricol*, vol. 20, no. 1, pp. 43–48, 2018.

[82] N. Vanetik and M. Litvak, “Definition extraction from generic and mathematical domains with deep ensemble learning,” *Mathematics*, vol. 9, no. 19, 2021, doi: 10.3390/math9192502.

[83] R. Guerizoli, “John Buridan on the Possibility of Defining Definition,” *History and Philosophy of Logic*, vol. 38, no. 3, pp. 201–209, 2017.

[84] D. T. H. Hutabarat, M. A. Efendi, M. Fatwa Str, and N. Prayoga, “Analyzing the Relationship Between Law and Technology,” *Policy, Law, Notary and Regulatory Issues (Polri)*, vol. 1, no. 2, pp. 99–110, 2022, doi: 10.55047/polri.v1i2.161.

[85] H. Xanthaki, “Emerging trends in legislation in Europe,” *Legislation in Europe. A comprehensive guide for scholars and practicioners. Hart, Oxford*, pp. 275–296, 2017.

[86] The European Parliament and the Council of the European Union, “Directive 2010/30/EU of the European Parliament and of the Council; on the indication by labelling and standard product information of the consumption of energy and other resources by energy-related products (recast),” *Official Journal of the European Union*, no. 4, pp. 1–12, 2010, [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32010L0030&from=EN

[87] D. Tiscornia and M. T. Sagri, “Legal Concepts and Multilingual Contexts in Digital Information,” *Beijing L. Rev.*, vol. 3, p. 73, 2012.

[88] K. Petress, “Critical Thinking: An Extended Definition,” *Education (Chula Vista)*, vol. 124, no. 3, p. 461, 2004, [Online]. Available: http://www.findarticles.com/p/articles/mi_qa3673/is_200404/ai_n9345203%3E%5Cnhttp://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=EJ698515&site=ehost-live%5Cnhttp://content.ebscohost.com/ContentServer.asp?T=P&P=AN&K=13186516&S=R&D=aph&EbscoConten

[89] J. Sanborn, *The Legal Aspects of Policing*. West Academic, 2018.

[90] Asiva Noor Rachmayani, “No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title,” p. 6, 2015.

[91] C. Sai, A. Damaratskaya, K. Winter, and S. Rinderle-Ma, “Identification and Visualization of Legal Definitions and Legal Term Relations,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14319 LNCS, pp. 151–161, 2023, doi: 10.1007/978-3-031-47112-4_14.

[92] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, “Environmental and Health Impacts of Air Pollution: A Review,” *Front Public Health*, vol. 8, no. February, pp. 1–13, 2020, doi: 10.3389/fpubh.2020.00014.

[93] J. Summers, “Tangible Intangibles in the United States’ Tax Cuts and Jobs Act: How Mixed Definitions of ‘Intangible’ Lead to Mixed Results in the United

States' Efforts to Close Tax Loopholes, Move to a Territorial Tax System, and Reduce Base Erosion and Profit Shifti," 2018.

[94] E. Westerhout, "Definition Extraction using Linguistic and Structural Features," in *Workshop On Definition Extraction 2009*, Borovets, Bulgaria, 2009, pp. 61–67.

[95] A. Turnaev and Z. Apanovich, "Extraction of the author's terminology and definitions from mathematical texts," *Bulletin of the Novosibirsk Computing Center, Computer Science series*, vol. 47, no. 47, pp. 43–51, 2024, doi: 10.31144/bncc.cs.2542-1972.2023.n47.p43-51.

[96] R. Stanković, C. Krstev, R. Stijović, M. Gočanin, and M. Škorić, "Towards automatic definition extraction for serbian," *EURALEX Proceedings*, vol. 2, pp. 695–703, 2021.

[97] S. Spala, N. A. Miller, Y. Yang, F. Dernoncourt, and C. Dockhorn, "DefT: A corpus for definition extraction in free- And semi-structured text," *LAW 2019 - 13th Linguistic Annotation Workshop, Proceedings of the Workshop*, pp. 124–131, 2019, doi: 10.18653/v1/w19-4015.

[98] L. Espinosa-Anke and S. Schockaert, "Syntactically aware neural architectures for definition extraction," *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, vol. 2, pp. 378–385, 2018, doi: 10.18653/v1/n18-2061.

[99] C. Borg, M. Rosner, and G. Pace, "Evolutionary Algorithms for Definition Extraction," *International Workshop on Definition Extraction, held in conjunction with the International Conference RANLP 2009 - Proceedings*, pp. 26–32, 2009.

[100] A. P. Ben Veyseh, F. Dernoncourt, D. Dou, and T. H. Nguyen, "A joint model for definition extraction with syntactic connection and semantic consistency," *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pp. 9098–9105, 2020, doi: 10.1609/aaai.v34i05.6444.

[101] P. Kumar, A. Singh, P. Kumar, and C. Kumar, "An explainable machine learning approach for definition extraction," in *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences*, 2020, pp. 145–155.

[102] R. H. Hwang, Y. L. Hsueh, and Y. T. Chang, "Building a Taiwan law ontology based on automatic legal definition extraction," *Applied System Innovation*, vol. 1, no. 3, pp. 1–20, 2018, doi: 10.3390/asi1030022.

[103] S. Höfler, A. Bünzli, and K. Sugisaki, "Detecting legal definitions for automated style checking in draft laws," *Technical Reports in Computational Linguistics*, no. CL-2011.01, 2011.

[104] T. Padayachy, B. Scholtz, and J. Wesson, "An information extraction model using a graph database to recommend the most applied case," in *2018*

International Conference on Computing, Electronics \& Communications Engineering (iCCECE), 2018, pp. 89–94.

[105] E. Ferneda, H. A. do Prado, A. H. Batista, and M. S. Pinheiro, "Extracting definitions from brazilian legal texts," in *International Conference on Computational Science and Its Applications*, 2012, pp. 631–646.

[106] L. Mommers and W. Voermans, "Using Legal Definitions to Increase the Accessibility of Legal Documents.," in *JURIX*, 2005, pp. 147–156.

[107] D. Bernsohn *et al.*, "LegalLens: Leveraging LLMs for Legal Violation Identification in Unstructured Text," *EACL 2024 - 18th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference*, vol. 1, pp. 2129–2145, 2024.

[108] L. Dini, W. Peters, D. Liebwald, E. Schweighofer, L. Mommers, and W. Voermans, "Cross-lingual legal information retrieval using a WordNet architecture," in *Proceedings of the 10th international conference on Artificial intelligence and law*, 2005, pp. 163–167.

[109] A. Flatt, A. Langner, and O. Leps, *Model-Driven Development of Akoma Ntoso Application Profiles*, January 2, 2023. Cham: Springer International Publishing, 2022. doi: 10.1007/978-3-031-14132-4.

[110] L. Weissweiler, V. Hofmann, M. J. Sabet, and H. Schütze, "CaMEL: Case Marker Extraction without Labels," Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.10010

[111] "Consequence Assessment in the Context of the ' Seveso Ii Directive '," vol. 2000, no. March 2002, pp. 193–230.

[112] K. Lacourse, B. Yetton, S. Mednick, and S. C. Warby, "Massive online data annotation, crowdsourcing to generate high quality sleep spindle annotations from EEG data," *Sci Data*, vol. 7, no. 1, pp. 1–14, 2020.

[113] R. Islamaj, D. Kwon, S. Kim, and Z. Lu, "TeamTat: a collaborative text annotation tool," *Nucleic Acids Res*, vol. 48, no. W1, pp. W5–W11, 2020.

[114] M. Asif, M. Bashir, M. A. Qureshi, H. M. Zain, and M. Shoaib, "Roman Urdu Sentiment Analysis of Reviews on PSL Anthems," vol. 06, no. 03, pp. 4–11, 2022.

[115] S. Archondakis, M. Roma, and E. Kaladelfou, "Remote cytological diagnosis of salivary gland lesions by means of precaptured videos," *J Am Soc Cytopathol*, 2021.

[116] Z. Chen and T. Qian, "Transfer capsule network for aspect level sentiment classification," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Italy, 2019, pp. 547–556.

[117] F. Dai, M. Leach, A. M. Macrae, M. Minero, and E. D. Costa, "Does thirty-minute standardised training improve the inter-observer reliability of the horse grimace scale (HGS)? A case study," *Animals*, vol. 10, no. 5, May 2020, doi: 10.3390/ani10050781.

[118] R. ul Haq, "EUR-Lex SDG-Annotated Dataset," 2025. [Online]. Available: https://huggingface.co/datasets/razaulhaq/eurlex-sdg-annotated

[119] G. Boella, L. Di Caro, and L. Humphreys, "Using classification to support legal knowledge engineers in the Eunomos legal document management system," *Fifth international workshop on Juris-informatics (JURISIN)*, 2011.

[120] O. M. Sulea, M. Zampieri, S. Malmasi, M. Vela, L. P. Dinu, and J. Van Genabith, "Exploring the use of text classification in the legal domain," *CEUR Workshop Proc*, vol. 2143, 2017.

[121] R. M. Palau and M. F. Moens, "Argumentation mining: The detection, classification and structure of arguments in text," *Proceedings of the International Conference on Artificial Intelligence and Law*, pp. 98–107, 2009, doi: 10.1145/1568234.1568246.

[122] N. Aletras, D. Tsarapatsanis, D. Preoţiuc-Pietro, and V. Lampos, "Predicting judicial decisions of the European court of human rights: A natural language processing perspective," *PeerJ Comput Sci*, vol. 2016, no. 10, pp. 1–19, 2016, doi: 10.7717/peerj-cs.93.

[123] I. Chalkidis, I. Androutsopoulos, and N. Aletras, "Neural legal judgment prediction in English," *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pp. 4317–4323, 2020, doi: 10.18653/v1/p19-1424.

[124] O. M. Şulea, M. Zampieri, M. Vela, and J. Van Genabith, "Predicting the law area and decisions of French supreme court cases," *International Conference Recent Advances in Natural Language Processing, RANLP*, vol. 2017-Septe, no. 2011, pp. 716–722, 2017, doi: 10.26615/978-954-452-049-6_092.

[125] T. S. Nguyen, L. M. Nguyen, S. Tojo, K. Satoh, and A. Shimazu, *Recurrent neural network-based models for recognizing requisite and effectuation parts in legal texts*, vol. 26, no. 2. Springer Netherlands, 2018. doi: 10.1007/s10506-018-9225-1.

[126] D. Ji, P. Tao, H. Fei, and Y. Ren, "An end-to-end joint model for evidence information extraction from court record document," *Inf Process Manag*, vol. 57, no. 6, 2020, doi: 10.1016/j.ipm.2020.102305.

[127] S. Undavia, A. Meyers, and J. E. Ortega, "A comparative study of classifying legal documents with neural networks," *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018*, vol. 15, pp. 515–522, 2018, doi: 10.15439/2018F227.

[128] H. Chen, L. Wu, J. Chen, W. Lu, and J. Ding, "A comparative study of automated legal text classification using random forests and deep learning," *Inf Process Manag*, vol. 59, no. 2, p. 102798, 2022, doi: 10.1016/j.ipm.2021.102798.

[129] C. J. Mahoney, "A Framework for Explainable Text Classification in Legal Document Review," *2019 IEEE International Conference on Big Data (Big Data)*, pp. 1858–1867, 2019.

[130] Q. Han and D. Snaidauf, "Comparison of Deep Learning Technologies in Legal Document Classification," *Proceedings - 2021 IEEE International Conference*

on Big Data, Big Data 2021, pp. 2701–2704, 2021, doi: 10.1109/BigData52589.2021.9671486.

[131] R. Keeling *et al.*, "Empirical Comparisons of CNN with Other Learning Algorithms for Text Classification in Legal Document Review," *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, pp. 2038–2042, 2019, doi: 10.1109/BigData47090.2019.9006248.

[132] I. Chalkidis, M. Fergadiotis, and I. Androutsopoulos, "MultiEURLEX - A multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer," *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pp. 6974–6996, 2021, doi: 10.18653/v1/2021.emnlp-main.559.

[133] N. Prasad, M. Boughanem, and T. Dkaki, "Exploring Large Language Models and Hierarchical Frameworks for Classification of Large Unstructured Legal Documents," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14609 LNCS, pp. 221–237, 2024, doi: 10.1007/978-3-031-56060-6_15.

[134] M. Qiu, Y. Zhang, T. Ma, Q. Wu, and F. Jin, "Convolutional-neural-network-based multilabel text classification for automatic discrimination of legal documents," *Sensors and Materials*, vol. 32, no. 8p2, pp. 2659–2672, 2020, doi: 10.18494/SAM.2020.2794.

[135] F. Wei, H. Qin, S. Ye, and H. Zhao, "Empirical Study of Deep Learning for Text Classification in Legal Document Review," *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, pp. 3317–3320, 2018, doi: 10.1109/BigData.2018.8622157.

[136] M. Y. Noguti, E. Vellasques, and L. S. Oliveira, "Legal Document Classification: An Application to Law Area Prediction of Petitions to Public Prosecution Service," *Proceedings of the International Joint Conference on Neural Networks*, no. 2, 2020, doi: 10.1109/IJCNN48605.2020.9207211.

[137] L. Wan, C. C. Llp, G. Papageorgiou, C. C. Llp, M. Seddon, and C. C. Llp, "Long-length Legal Document Classification," 2014.

[138] and J. A. André Aguiar, Raquel Silveira, Vládia Pinheiro, Vasco Furtado and Neto, "Text Classification in Legal Documents Extracted from Lawsuits in Brazilian Courts," in *Intelligent Systems*, Intelligent Systems, Springer, 2021, pp. 586–600.

[139] P. H. L. De Araujo, T. E. De Campos, F. A. Braz, and N. C. Silva, "VICTOR : a dataset for Brazilian legal documents classification," no. May, pp. 1449–1458, 2020.

[140] L. Pukelis, N. B. Puig, M. Skrynik, and V. Stanciauskas, "OSDG-Open-Source Approach to Classify Text Data by UN Sustainable Development Goals (SDGs)."

[141] L. Pukelis, N. Bautista-Puig†, G. Statulevičiūtė, V. Stančiauskas, G. Dikmener✺, and D. Akylbekova✺, "OSDG 2.0: a multilingual tool for classifying text data by UN Sustainable Development Goals (SDGs)."

[142] J. E. Guisiano, R. Chiky, and J. De Mello, “SDG-Meter : a deep learning based tool for automatic text classification of the Sustainable Development Goals.” [Online]. Available: https://sdg-tracker.org

[143] R. C. Morales-Hernández, D. Becerra-Alonso, E. R. Vivas, and J. Gutiérrez, “Comparison Between SVM and DistilBERT for Multi-label Text Classification of Scientific Papers Aligned with Sustainable Development Goals,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13613 LNAI, pp. 57–67, 2022, doi: 10.1007/978-3-031-19496-2_5.

[144] J. Lee *et al.*, “Machine Learning Driven Aid Classification for Sustainable Development,” pp. 6040–6048, 2023, doi: 10.24963/ijcai.2023/670.

[145] R. C. Morales-Hernández, J. G. Juagüey, and D. Becerra-Alonso, “A Comparison of Multi-Label Text Classification Models in Research Articles Labeled with Sustainable Development Goals,” *IEEE Access*, vol. 10, no. November, pp. 123534–123548, 2022, doi: 10.1109/ACCESS.2022.3223094.

[146] A. Joshi *et al.*, “A Knowledge Organization System for the United Nations Sustainable Development Goals,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 548–564. doi: 10.1007/978-3-030-77385-4_33.

[147] A. Pincet, S. Okabe, and M. Pawelczyk, “Linking aid to the Sustainable Development Goals - a machine learning approach,” 2019.

[148] S. Sato, T. Hashimoto, and Y. Shirota, “Evaluation for ESD (Education for Sustainable Development) to achieve SDGs at University,” *2020 11th International Conference on Awareness Science and Technology, iCAST 2020*, 2020, doi: 10.1109/iCAST51195.2020.9319406.

[149] I. Mandilara, E. Fotopoulou, C. M. Androna, A. Zafeiropoulos, and S. Papavassiliou, “Knowledge Graph Data Enrichment Based on a Software Library for Text Mapping to the Sustainable Development Goals,” *CEUR Workshop Proc*, vol. 3447, no. May, pp. 51–69, 2023.

[150] D. Devi, S. K. Biswas, and B. Purkayastha, “A Review on Solution to Class Imbalance Problem: Undersampling Approaches,” in *2020 International Conference on Computational Performance Evaluation (ComPE)*, 2020, pp. 626–631.

[151] Z. Wu, Z. Wang, J. Chen, H. You, M. Yan, and L. Wang, “Stratified random sampling for neural network test input selection,” *Inf Softw Technol*, vol. 165, no. February 2023, p. 107331, 2024, doi: 10.1016/j.infsof.2023.107331.

[152] M. A. Qureshi *et al.*, “Sentiment Analysis of Reviews in Natural Language: Roman Urdu as a Case Study,” *IEEE Access*, vol. 10, no. 1, pp. 24945–24954, 2022, doi: 10.1109/ACCESS.2022.3150172.

[153] D. J. Kalita, V. P. Singh, and V. Kumar, “A Survey on SVM Hyper-Parameters Optimization Techniques,” in *Social Networking and Computational Intelligence*, Springer, 2020, pp. 243–256.

[154] M. L. Zhang and Z. H. Zhou, “ML-KNN: A lazy learning approach to multi-label learning,” *Pattern Recognit*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007, doi: 10.1016/j.patcog.2006.12.019.

[155] P. Soucy and G. W. Mineau, “A simple KNN algorithm for text categorization,” in *Proceedings - IEEE International Conference on Data Mining, ICDM*, IEEE Comput. Soc, 2001, pp. 647–648. doi: 10.1109/icdm.2001.989592.

[156] C. Niculita and L. Dumitriu, “An Experiment on Text Summarization: Frequent Terms and Concept Definition Extraction,” *2020 24th International Conference on System Theory, Control and Computing, ICSTCC 2020 - Proceedings*, pp. 78–83, 2020, doi: 10.1109/ICSTCC50638.2020.9259708.

[157] M. Mhatre, D. Phondekar, P. Kadam, A. Chawathe, and K. Ghag, “Dimensionality reduction for sentiment analysis using pre-processing techniques,” in *Proceedings of the International Conference on Computing Methodologies and Communication, ICCMC 2017*, IEEE, Jul. 2018, pp. 16–21. doi: 10.1109/ICCMC.2017.8282676.

[158] M. Mhatre, D. Phondekar, P. Kadam, A. Chawathe, and K. Ghag, “Dimensionality reduction for sentiment analysis using pre-processing techniques,” in *Proceedings of the International Conference on Computing Methodologies and Communication, ICCMC 2017*, 2018, pp. 16–21. doi: 10.1109/ICCMC.2017.8282676.

Appendix

Table 25: The Frequency of each SDGs Goal in the EU Legislative files by taking the first four articles' text as input data

SDG Goals	Frequency in Legislative Files
1	240
2	110
3	248
4	63
5	8
6	22
7	135
8	436
9	352
10	305
11	120
12	129
13	88
14	84
15	69
16	579
17	183

Table 26: The Frequency of each SDGs Goal in the EU Legislative files by taking all the articles' text as input data

SDG Goals	Frequency in Legislative Files
1	241
2	109
3	219
4	66
5	11
6	16
7	138
8	420

9	323
10	297
11	103
12	122
13	93
14	88
15	69
16	585
17	193

Table 27: The Frequency of each SDGs Goal in the EU Legislative files by taking the text of the first four articles' with the preambles text as input data

SDG Goals	Frequency in Legislative Files
1	299
2	150
3	332
4	87
5	14
6	40
7	190
8	497
9	422
10	398
11	149
12	199
13	136
14	115
15	133
16	610
17	258

Table 28: The Frequency of each SDGs Goal in the EU Legislative files by taking all the articles' text with the preambles text as input data

SDG Goals	**Frequency in Legislative Files**
1	290

2	149
3	304
4	83
5	16
6	23
7	184
8	484
9	415
10	368
11	134
12	183
13	136
14	109
15	128
16	608
17	249

Table 29: The Frequency of each SDGs Goal in the EU Legislative files by taking all the text of each file as input data

SDG Goals	Frequency in Legislative Files
1	297
2	157
3	326
4	77
5	15
6	30
7	217
8	497
9	476
10	381
11	173
12	193
13	140
14	108
15	135
16	677
17	245

Table 30: The Frequency of each SDGs Goal in the EU Legislative files by taking all the Delimiting Definition as input data

SDG Goals	Frequency in Legislative Files
1	265
2	297
3	700
4	25
5	12
6	10
7	217
8	1278
9	689
10	343
11	153
12	122
13	122
14	59
15	11
16	2910
17	220

Table 31: The Frequency of each SDGs Target in the EU Legislative files by taking the first four articles' text as input data

SDGs Targets	**Frequency in Legislative Files**	**SDGS Targets**	**Frequency in Legislative Files**	**SDGs Targets**	**Frequency in Legislative Files**
1.1	1	7.3	86	13.2	72
1.2	26	8.1	106	14.1	4
1.3	77	8.2	73	14.2	25
1.4	66	8.3	37	14.3	39
1.5	10	8.4	2	14.4	67
2.1	55	8.5	22	14.5	17
2.2	20	8.6	1	14.6	47

2.3	46	8.7	4	15.1	2
2.4	5	8.8	35	15.2	32
2.5	15	9.1	58	15.3	11
3.1	1	9.2	5	15.5	31
3.2	22	9.3	77	15.7	1
3.3	10	9.4	25	15.8	8
3.4	113	9.5	101	15.9	3
3.5	14	10.1	1	16.1	266
3.7	12	10.2	12	16.2	173
3.8	41	10.3	124	16.3	150
3.9	3	10.4	1	16.4	316
4.1	5	10.5	7	16.5	71
4.2	2	10.7	79	16.6	35
4.3	8	11.1	26	16.7	9
4.4	27	11.2	70	16.9	39
4.6	3	11.3	5	17.1	101
5.1	1	11.4	14	17.2	1
5.2	12	11.6	6	17.5	1
5.3	7	11.7	1	17.6	6
5.4	1	12.2	9	17.8	1
6.1	12	12.3	9	17.9	4
6.2	4	12.4	20	17.12	1
6.3	10	12.5	75	17.13	1
6.5	1	12.6	1	17.14	3
7.1	44	12.8	5	17.17	4
7.2	33	13.1	16	17.19	2

Table 32: The Frequency of each SDGs Target in the EU Legislative files by taking all the articles' text as input data

SDGs Targets	Frequency in Legislative Files	SDGS Targets	Frequency in Legislative Files	SDGs Targets	Frequency in Legislative Files
1.1	1	7.3	89	14.1	3
1.2	39	8.1	129	14.2	26
1.3	86	8.2	69	14.3	52
1.4	72	8.3	55	14.4	69
1.5	13	8.4	2	14.5	27
2.1	52	8.5	29	14.6	58
2.2	32	8.6	3	15.1	2
2.3	55	8.7	6	15.2	39
2.4	8	8.8	47	15.3	14
2.5	15	9.1	64	15.5	46
3.1	1	9.2	7	15.6	2
3.2	27	9.3	79	15.7	1
3.3	12	9.4	24	15.8	13
3.4	135	9.5	105	15.9	3
3.5	17	10.1	1	16.1	293
3.7	15	10.2	21	16.2	183
3.8	47	10.3	145	16.3	178
3.9	3	10.4	1	16.4	350
4.1	4	10.5	8	16.5	90
4.2	3	10.7	99	16.6	48
4.3	14	11.1	32	16.7	10
4.4	36	11.2	66	16.9	51
4.5	1	11.3	6	17.1	131
4.6	6	11.4	14	17.2	2

5.1	1	11.5	1	17.5	1
5.2	17	11.6	8	17.6	6
5.3	8	11.7	1	17.8	1
5.4	1	12.2	11	17.9	5
5.6	1	12.3	10	17.12	1
6.1	24	12.4	24	17.13	2
6.2	4	12.5	85	17.14	4
6.3	13	12.6	1	17.17	4
6.5	1	12.8	5	17.18	3
7.1	46	13.1	19	17.19	5
7.2	37	13.2	81		

Table 33: The Frequency of each SDGs Target in the EU Legislative files by taking the text of the first four articles' with preamble text as input data

SDGs Targets	Frequency in Legislative Files	SDGS Targets	Frequency in Legislative Files	SDGs Targets	Frequency in Legislative Files
1.1	1	7.3	134	14.1	4
1.2	37	8.1	148	14.2	55
1.3	116	8.2	107	14.3	58
1.4	99	8.3	80	14.4	74
1.5	18	8.4	2	14.5	58
2.1	79	8.5	45	14.6	80
2.2	37	8.6	2	15.1	3
2.3	65	8.7	5	15.2	53
2.4	15	8.8	70	15.3	22
2.5	22	9.1	102	15.5	73
3.1	1	9.2	17	15.6	1

3.2	28	9.3	103	15.7	1
3.3	33	9.4	35	15.8	14
3.4	170	9.5	146	15.9	3
3.5	28	10.1	1	16.1	318
3.7	16	10.2	26	16.2	203
3.8	71	10.3	181	16.3	188
3.9	4	10.4	1	16.4	351
4.1	5	10.5	15	16.5	101
4.2	3	10.7	97	16.6	57
4.3	15	11.1	37	16.7	12
4.4	47	11.2	83	16.9	54
4.5	1	11.3	6	17.1	175
4.6	7	11.4	15	17.2	2
5.1	1	11.5	1	17.5	2
5.2	10	11.6	11	17.6	8
5.3	9	11.7	1	17.8	1
5.4	1	12.2	12	17.9	5
5.6	1	12.3	10	17.12	1
6.1	14	12.4	31	17.13	4
6.2	4	12.5	127	17.14	4
6.3	16	12.6	1	17.17	5
6.5	1	12.8	5	17.18	2
7.1	67	13.1	31	17.19	6
7.2	56	13.2	113		

Table 34: The Frequency of each SDGs Target in the EU Legislative files by taking the text of all the articles' with preamble text as input

data

SDGs Targets	Frequency in Legislative Files	SDGS Targets	Frequency in Legislative Files	SDGs Targets	Frequency in Legislative Files
1.1	1	7.3	137	14.1	4
1.2	47	8.1	154	14.2	51
1.3	118	8.2	102	14.3	62
1.4	94	8.3	81	14.4	72
1.5	15	8.4	2	14.5	49
2.1	76	8.5	48	14.6	77
2.2	40	8.6	4	15.1	2
2.3	67	8.7	5	15.2	52
2.4	12	8.8	68	15.3	20
2.5	19	9.1	95	15.5	67
3.1	1	9.2	16	15.6	2
3.2	26	9.3	96	15.7	1
3.3	23	9.4	31	15.8	15
3.4	177	9.5	135	15.9	3
3.5	29	10.1	1	16.1	322
3.7	17	10.2	29	16.2	204
3.8	61	10.3	183	16.3	207
3.9	3	10.4	1	16.4	361
4.1	5	10.5	16	16.5	106
4.2	3	10.7	106	16.6	58
4.3	18	11.1	36	16.7	12
4.4	46	11.2	79	16.9	57
4.5	1	11.3	7	17.1	187

4.6	6	11.4	15	17.2	2
5.1	1	11.5	1	17.5	2
5.2	16	11.6	10	17.6	9
5.3	8	11.7	1	17.8	1
5.4	1	12.2	11	17.9	5
5.6	1	12.3	10	17.12	2
6.1	25	12.4	29	17.13	4
6.2	5	12.5	118	17.14	4
6.3	15	12.6	1	17.17	5
6.5	1	12.8	5	17.18	3
7.1	65	13.1	29	17.19	7
7.2	53	13.2	114		

Table 35: The Frequency of each SDGs Target in the EU Legislative files by taking the whole text of each file as input data

SDGs Targets	Frequency in Legislative Files	SDGS Targets	Frequency in Legislative Files	SDGs Targets	Frequency in Legislative Files
1.1	1	7.3	160	13.2	122
1.2	42	8.1	142	14.1	4
1.3	108	8.2	108	14.2	42
1.4	96	8.3	65	14.3	54
1.5	16	8.4	2	14.4	72
2.1	77	8.5	38	14.5	52
2.2	37	8.6	3	14.6	70
2.3	70	8.7	5	15.1	3
2.4	11	8.8	64	15.2	46
2.5	21	9.1	102	15.3	20
3.1	1	9.2	12	15.5	63

3.2	25	9.3	88	15.6	1
3.3	28	9.4	65	15.7	1
3.4	179	9.5	137	15.8	13
3.5	27	10.1	1	15.9	3
3.7	14	10.2	24	16.1	297
3.8	65	10.3	165	16.2	191
3.9	4	10.4	1	16.3	190
4.1	5	10.5	14	16.4	335
4.2	3	10.7	105	16.5	87
4.3	15	11.1	39	16.6	43
4.4	47	11.2	91	16.7	13
4.5	1	11.3	5	16.9	53
4.6	6	11.4	15	17.1	170
5.1	1	11.5	1	17.2	2
5.2	14	11.6	10	17.5	2
5.3	8	11.7	1	17.6	13
5.4	1	12.2	10	17.8	1
5.6	1	12.3	9	17.9	5
6.1	26	12.4	30	17.12	2
6.2	5	12.5	121	17.13	3
6.3	13	12.6	1	17.14	6
6.5	1	12.7	1	17.17	5
7.1	68	12.8	5	17.18	2
7.2	51	13.1	23	17.19	6

Table 36: The Frequency of SDGs Target in Delimiting Definition

SDGs Targets	Frequency in Legislative Files	SDGS Targets	Frequency in Legislative Files	SDGs Targets	Frequency in Legislative Files
1.3	86	8.3	50	13.2	107
1.4	77	8.4	5	14.1	16
1.5	5	8.5	4	14.2	31
2.1	125	8.6	1	14.3	80
2.2	5	8.7	1	14.4	62
2.3	38	8.8	5	14.5	11
2.5	14	9.1	41	14.6	28
3.2	43	9.3	88	15.1	1
3.3	7	9.4	7	15.2	18
3.4	175	9.5	45	15.3	5
3.5	2	10.2	6	15.8	6
3.7	45	10.3	81	15.9	1
3.8	39	10.5	66	16.1	280
3.9	10	10.7	24	16.2	150
4.1	2	11.1	21	16.3	145
4.2	1	11.2	164	16.4	999
5.2	16	11.3	4	16.5	74
5.3	6	11.4	9	16.6	15
6.1	2	11.6	9	16.7	5
6.3	9	12.2	5	16.9	14
7.1	69	12.3	7	17.1	122
7.2	94	12.4	26	17.6	3
7.3	106	12.5	87	17.9	2
8.1	239	12.8	4	17.17	2
8.2	74	13.1	2		

Borsa di dottorato del Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 (CCI 2014IT16M2OP005), risorse FSE REACT-EU, Azione IV.4 “Dottorati e contratti di ricerca su tematiche dell’innovazione” e Azione IV.5 “Dottorati su tematiche Green" J35F21003170006.