Alma Mater Studiorum – Università di Bologna
in partnership with LAST-JD Consortium
Università degli studi di Torino
Universitat Autonoma de Barcelona
Mykolas Romeris University
Tilburg University
and in cotutorship with the
The University of Luxembourg

PhD Programme in

Erasmus Mundus Joint International Doctoral Degree in Law, Science and Technology
Cycle 28 – a.y. 2012/13

Settore Concorsuale di afferenza: 12H3

Settore Scientifico disciplinare: IUS20

(Title of the Thesis)

A COMBINED UNSUPERVISED TECHNIQUE FOR AUTOMATIC CLASSIFICATION IN ELECTRONIC DISCOVERY

Submitted by: ENIAFE FESTUS AYETIRAN

The PhD Programme Coordinator	Supervisor (s)
Prof. Monica Palmirani	
	Prof. Guido Boella Prof. Leon van der Torre Dr. Luigi Di Caro

Year 2017

PhD-FSTC-2017-01

The Faculty of Sciences, Technology and Communication

University of Bologna
Law School

DISSERTATION

Defence held on 31/01/2017 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

ENIAFE FESTUS AYETIRAN
Born on 9th January, 1984 in Ilutitun (Nigeria)

A Combined Unsupervised Technique for Automatic Classification in Electronic Discovery

Dissertation defence committee

Dr Leon van der Torre, dissertation supervisor
Professor, Université du Luxembourg

Dr Guido Boella, dissertation supervisor
Professor, Università degli Studi di Torino

Dr Andrea Maurino, Chairman
Professor, Università degli Studi di Milano-Bicocca

Dr Schweighofer Erich, Vice-chairman
Professor, University of Vienna

Dr Monica Palmirani
Professor, Università degli Studi di Bologna

Dr Cataldi Mario
Professor, University of Paris8

Declaration

I hereby declare that the content of this thesis is my original work except where adequate references have been made to the work of others.

AYETIRAN Eniafe Festus
December 2016

Acknowledgements

First and formost, I thank the almighty God - the giver everything, for the grace to commence and complete this programme. My special thanks goes to the LAST-JD team: the coordinator, Prof. Monica Palmirani, the doctoral board members and the academic committee. Special thanks to Dina Ferarri and Tazia Bianchi for all their efforts, both administratively and personally. My appreciation goes to my supervisors, Professors Guido Boella and Leon van der Torre and Dr Luigi Di Caro.

Many thanks to my colleagues for memorable times spent together in Bologna, Turin, Barcelona and Luxembourg and for their inspirations. I specially thank my parents; Mrs and Mrs. Ayetiran, my siblings, friends and well-wishers for their support both financially and morally when the entire journey started from elementary school up till this time. I acknowedge the support of European Commission without whose scholarship, this thesis wouldn't have been possible.

Finally, special thanks to my wife, Oluwafemi and my kids, Alexander and Daniel for being there while I was away on this journey. You are all great!

Abstract

Electronic data discovery (EDD), e-discovery or eDiscovery is any process by which electronically stored information (ESI) is sought, identified, collected, preserved, secured, processed, searched for the ones relevant to civil and/or criminal litigations or regulatory matters with the intention of using them as evidence. Searching electronic document collections for relevant documents is part of eDiscovery which poses serious problems for lawyers and their clients alike. Getting efficient and effective techniques for search in eDiscovery is an interesting and still an open problem in the field of legal information systems. Researchers are shifting away from traditional keyword search to more intelligent approaches such as machine learning (ML) techniques. State-of-the-art algorithms for search in eDiscovery focus mainly on supervised approaches, mainly; supervised learning and interactive approaches. The former uses labelled examples for training systems while the latter uses human assistance in the search process to assist in retrieving relevant documents. Techniques in the latter approach include interactive query expansion among others. Both approaches are supervised form of technology assisted review (TAR). Technology assisted review is the use of technology to assist or completely automate the process of searching and retrieval of relevant documents from electronically stored information (ESI). In text retrieval/classification, supervised systems are known for their superior performance over unsupervised systems. However, two serious issues limit their application in the electronic discovery search and information retrieval (IR) in general. First, they have associated high cost in terms of finance and human effort. This is particularly responsible for the huge amount of money expended on eDiscovery on annual basis. Secondly, their case/project-specific nature does not allow for resuse, thereby contributing more to organizations' expenses when they have two or more cases involving eDiscovery.

Unsupervised systems on the other hand, is cost-effective in terms of finance and human effort. A major challenge in unsupervised ad hoc information retrieval is that of vocabulary problem which causes terms mismatch in queries and documents. While topic modelling techniques try to tackle this from the thematic point of view in the sense that both queries and documents are likely to match if they discuss about the same topic, natural language processing (NLP) approaches view it from the semantic perspective. Scalable topic modelling algorithms, just like the traditional bag of words technique, suffer from polysemy and synonymy problems. Natural language processing techniques on the other hand, while being able to considerably resolve the polysemy and synonymy problems are computationally expensive and not suitable for large collections as is the case in eDiscovery. In this thesis, we exploited the peculiarity of eDiscovery collections being composed mainly of e-mail communications and their attachments, mining topics of discourse from e-mails and disambiguating these topics and queries for terms matching has been proven to be effective for retrieving relevant documents when compared to traditional stem-based retrieval.

In this work, we present an automated unsupervised approach for retrieval/classification in eDiscovery. This approach is an ad hoc retrieval which creates a representative for each original document in the collection using latent dirichlet allocation (LDA) model with Gibbs sampling and explores word sense disambiguation (WSD) to give these representative documents and queries deeper meanings for distributional semantic similarity. The word sense disambiguation technique by itself is a hybrid algorithm derived from the modified version of the original Lesk algorithm and the Jiang & Conrath similarity measure.

Evaluation was carried out on this technique using the TREC legal track. Results and observations are discussed in chapter 8. We conclude that WSD can improve ad hoc retrieval effectiveness. Finally, we suggest further on efficient algorithms for word sense disambiguation which can further improve retrieval effectiveness if applied to original document collections against using representative collections.

Table of contents

List of figures

List of tables

Chapter 1

Background and General Introduction

1.1 Introduction

With millions of electronically stored information (ESI) in organizations' repositories nowadays, retrieving the relevant ones to a case in response to production requests from an opposing party during litigations, is a difficult task for both organizations and their lawyers.

Electronic data discovery (EDD), e-discovery or eDiscovery is any process by which electronically stored information (ESI) is sought, identified, collected, preserved, secured, processed, and searched for the ones relevant to civil and/or criminal litigations or regulatory matters with the intention of using them as evidence. It came into limelight as a result of the amendment made to the United States federal rules of civil procedure which entered into effect on December 1, 2006. Another amendment was proposed in December 1 2014 but the one which makes E-discovery an important focus area is the 2006 amendment. The federal rules of civil procedure (FRCP) are a set of rules focused on governing court procedures for managing civil suits in the United States district courts. While the United States Supreme Court is responsible for creating and managing FRCP, the United States Congress must approve these rules and any changes made to them. The specific rules aimed at electronically stored information are rules 16, 23, 26, 33, 34, 37 and 45. E-discovery (or E-disclosure as the case may be) is also practised in other jurisdictions such as the United Kingdom, Australia, Hong Kong, Canada etc. However, even individuals or organizations from countries that do not practice E-discovery may still find themselves subjected to E-discovery if they have cases or trade with other parties in countries where E-discovery is being practised.

In a litigation or regulatory matter, a party to the case requests some set of electronic documents meeting certain criteria from the other party in order to tender these documents as evidence in court. This formal request, referred to as request for production (or production request) lists some criteria and conditions the electronic documents to be produced must meet. After production, the requesting party manually reviews the produced documents for relevance. As a result of penalties associated with proven cases of deliberate withholding of documents which are not subject of privilege information, the discovery team of the producing party tries as much as possible to produce enough documents deemed to meet the information needs of the requesting party. Discovering and producing the required document(s) among huge volume of data created and stored electronically in various formats in repositories is a big challenge which needs to be addressed with improvement in its current state. The search for efficient and effective search/classification algorithms in eDiscovery is still an open problem in the field of legal information systems. eDiscovery, in contrast to some other forms of information retrieval (IR) has a peculiar characteristic in that it is a classification task precisely text categorization, in which there are two main categories; either a document is relevent or not relevant. It therefore employs information retrieval techniques to do this categorization.

Approaches to search/classification for eDiscovery can be manual or technology-assisted, supervised or unsupervised, fully automated or combination of both manual and technology. However, an eDiscovery search algorithm can combine features of the different approaches. For instance, a system may be fully automated but employs supervised or unsupervised techniques in its development. Furthermore, an interactive system may use an automated and/or learning techniques for the technology aspect while using manual interaction during search. For instance, some systems for the interactive task of the 2010 TREC legal track [22] make use of supervised techniques. Manual approach engages human effort to exhaustively search a collection for electronic documents meeting a request for production (topic) while technology-assisted review involves the use of technology to assist or completely automate the process of retrieving relevant documents from a collection. Furthermore, techniques for technology-assisted system may be fully automated or employs human and technology (e.g interactive search) in its operation. Also, a technology-assisted systems may employ supervised or unsupervised techniques in their development.

We generally refer to any approach that employs human effort in the search and classification process loop as supervised, of which the interactive task is an example. We distinguish this from automated machine learning techniques. For instance, a system may be trained to learn and work based on labelled examples (training datasets) but does not need manual intervention in its operation. In the case of eDiscovery, these examples are documents which have been labelled as either relevant or not relevant by the person training the system. On the other hand, unsupervised machine learning techniques do not use labelled examples but learn from the test dataset itself. The cost of developing training set is one that is very expensive and difficult especially in huge collections. In addition to this shortcoming, it also presents an unfortunate situation in the sense that it does not allow for reuse of the system trained on this training set on other collections and/or production requests. For the rest of this thesis, we refer to supervised technology-assisted review approaches as encompassing both systems using supervised machine learning algorithms or interactive approaches i.e using humans in the search process.

It has been reported that technology assisted review in eDiscovery can also be more effective and more efficient than exhaustive manual review [39]. But the big issues are: (i) how much of human effort in technology assisted review techniques can vividly distinguish them from exhaustive manual review and (ii) Does the outcomes justify the costs. The ultimate gain in technology-assisted review is the ability to spend less in terms of human effort and money while still achieving an acceptable degree of effectiveness and efficiency. Supervised machine learning approaches, for instance, suffer from their case-specific nature which does not allow for reuse. It can be interesting to develop a system, give specific instructions to the system with specified information need without having to intervene in the working process of the system or having to laboriously develop training dataset, do rigorous training while at the same time being able to reuse the system irrespective of the case or the dataset involved. This description is a form of ad hoc information retrieval; a standard retrieval task in which the user specifies his information need through a query and initiates a search (executed by the information retrieval system) for documents which are likely to meet the information need of the user. Supervised systems lack these features. In this thesis, we investigate the application of an unsupervised and automated ad hoc retrieval technique for search/classification in eDiscovery. It investigates, on one hand, the effectiveness of sense-based ad hoc retrieval. On the other hand, it mitigates the efficiency problem associated with word sense disambiguation by using topical representations of documents in collections.

1.2 Evolution of Unsupervised Ad hoc Information Retrieval Technologies for eDiscovery

Keywords search has been the major standard for searching collections for discovery documents besides manual search until recently. Keywords search basically involves identifying and retrieving documents whose contents match the terms in a query. The results from this exercise are far from what can be referred to as meeting the information needs of users due to high degree of false hits. This is due mainly to the vocabulary problem of keyword search caused by polysemy/synonymy issues. Polysemy refers to words that have the same lexical spellings but different meanings. Using a keyword search will directly match polysemous words found in both query and documents without taking into consideration their meanings thereby contributing to degradation in the effectivesness of search results. Synonymy on the other hand, refers to words having different lexical spellings but share the same meaning. In relation to keywords search, synonymy contributes to query-document mismatch even when the terms in both query and documents have semantic similarity since terms matching uses spelling. In general, prevalence of synonymy in documents decreases the recall in retrieval systems while polysemy leads to decreased precision. Although keywords search have been adapted in other modified methods which includes iterative search in the form interactive query expansion. This works by agreeing on a specified search terms and repetitively adjusting and/or expanding the terms based on the initial results obtained. The current state of this approach is still short of meeting users' information needs. Automatic attempts to deal with the synonymy problem have relied on automatic term expansion using lexicons or the construction of a thesaurus. They are presumably advantageous to help improve recall, however, the drawback for fully automatic methods is that some added terms may have different meaning from the intended meaning thereby leading to performance degradation of precision of search results with no significant improvement in recall. Therefore for an effective automated system, there is need for accuracy in the choice of word sense of the terms to be expanded. In other words, ambiguity in query terms needs to be removed for improved precision to be achieved.

Analytic search does not rely entirely on the keywords but attempts to retrieve documents based on the keywords and the content of the documents. The basis for this search technology is the conversion of documents' contents into numeric values that allows the computer to compare different documents' values in order to determine similarity of content. This is advantageous in that if offers dimensionality reduction over the bag of words representation.

Latent semantic indexing (LSI), also called latent semantic analysis (LSA) [25] is a theory and method for extracting and representing the contextual meaning of words by statistical computations applied to a large corpus of text. The approach takes advantage of implicit higher order structure in the association of terms with documents (called the "semantic structure") in order to improve the detection of relevant documents on the basis of terms found in queries. The particular technique used is the singular value decomposition, in which a large term-by-document matrix is decomposed into a set of orthonormal factors from which the original matrix can be approximated by linear combination. The main aim of LSI is to retrieve related documents matching the query terms by doing analysis on the collection to obtain related documents along with those matching the query terms even though they do not have direct match with the query terms. This technique was widely reported in the literature to largely tackle the problem of synonymy and polysemy with improved performance over traditional keywords search but a major issue with this technique is that of scalability. According to Manning et al. [66], "even for a collection of modest size, the term-document matrix C is likely to have several tens of thousand of rows and columns, and a rank in the tens of thousands as well". They concluded with the following points:

- The computational cost of the singular vector decomposition is enormous. They stated that at the time of writing their material, they were not aware of any successful experiment with over one million documents. This constitutes the biggest obstacle to the widespread adoption to LSI.

- As the subscript, k of the matrix C_k is reduced, recall tends to increase.

- A value of k in the low hundreds can actually increase precision on some query benchmarks. This tends to suggest that for a suitable value of k, LSI addresses some of the problems of synonymy.

- LSI works best in applications where there is little overlap between queries and documents.

All the identified drawbacks limit the successful application of this technique in eDiscovery which often involves millions of electronic documents with the possibility of significant overlap.

The probabilistic latent semantic indexing (pLSI) model, also known as the aspect model [44], an alternative to and improvement on LSI, is a novel statistical technique for the analysis of two-mode and co-occurrence data, which has applications in information retrieval and filtering, natural language processing, machine learning from text, and in related areas. It is one of the first statistical and generative topic model developed from a training corpus of text documents by a generalization of the expectation maximization algorithm and believed to be able to deal with domain-specific synonymy problem as well as with polysemous words. pLSI is not a well-defined generative model of documents; there is no natural way to use it to assign probability to a previously unseen document. Another problem with pLSI, which also stems from the use of a distribution indexed by training documents, is that the number of parameters which must be estimated grows linearly with the number of training documents [11]. The parameters for a k-topic using pLSI model are k multinomial distributions of size V and M mixtures over the k hidden topics. This gives $kV + kM$ parameters and therefore linear growth in M. The linear growth in parameters suggests that the model is prone to overfitting. Empirically, overfitting has been found prone to serious problem [11]. Usually, a normalizing heuristic is used to smooth the parameters of the model for acceptable predictive performance. It has been shown, however, that overfitting can occur even when normalization is used [82]. Just like LSI, PLSI is also highly computationally intensive.

The latent dirichlet allocation (LDA) model [11] (see section 4.1.1) is arguably one of the most popular of the probabilistic analytic models. It is one of the main constituents on which this work is based.

1.3 Motivation for the Study

Despite the fact that so much research effort has been concentrated towards efficient and effective search techniques for eDiscovery over the years, the cost of eDiscovery continues to rise. The bulk of this cost is on conducting search. In 2014, it is estimated that the enterprise eDiscovery software marketplace was \$1.8 billion in total software revenue worldwide, an increase of 10.6% from 2013 [111] with several times that amount usually spent on the deployment, staffing and processing costs to effectively use the systems [77]. The major reason for this huge cost is attributable to organizations' engagement in project-based approach to conducting search in eDiscovery. This, in other words, borders on the case-specificity of supervised systems which they often embark upon. Supervised systems are known for good performance, they however come with very huge cost and and not suitable for reuse. According to the 2015 Gartner report [111], the following are changes in the eDiscovery business of organizations in contrast to previous years:

- Migration to Office 365 has kick started evaluation discussions and upgrade projects on eDiscovery processes and tools. There is need to reflect on what this meant to already established eDiscovery processes and technology applications.

- New data sources and increasing concerns about data sovereignty are driving emerging requirements and expanding the consideration of eDiscovery scope and technology usage. Organizations have started the dialogue on how to preserve social, web and internet of things data. This has raised the following fundamental questions: (i) Should these new types of content be subject to the same data preservation rules? and (ii) Could existing technologies support these new content types?

- Organizations want agile and less expensive approaches to eDiscovery. Many organizations have realized that the traditional project-based approach to eDiscovery are, in many ways, becoming unsustainable. This awareness motivates organizations to seek newer and innovative technologies that support lower cost and faster performance.

- Pricing structure continues to be simplified. The business goal of controlling eDiscovery costs by corporations has pushed the emphasis on price prediction and transparency. Many established vendors have been reducing the complexity in their pricing structure and newer vendors, especially the cloud native vendors, have offered much simpler pricing options. This factor, along with competition, is driving the eDiscovery costs to be more comparable and more cost-effective.

- Vendors increasingly expand their offering by deploying their eDiscovery software in the Software-as-a-Service (SaaS) model. Some of these SaaS services are offered in cloud. In addition to the benefits of cloud economics and scalability, eDiscovery in the cloud is becoming an appealing option if the data source resides in the cloud. This is a new area for eDiscovery practitioners.

- Another round of software market for eDiscovery is looming with the entrance of other big software vendors in eDiscovery market. For example, Microsoft has entered the eDiscovery software market by acquiring Equivio, kCura has expanded its relativity platform to collection and processing, and a handful of startups e.g Everlaw and Zapproved are steadily gaining customers.

This research has been motivated by the aforementioned changes and associated challenges especially the ones which borders on cost-effectiveness and performance. In fact, a lot organizations with good cases have been forced to opt for "out-of-court" settlements due to inability to afford eDiscovery cost. As a result, there is need to focus more research on effective, general purpose techniques for search and classification in eDiscovery with a considerable reduction in cost that can be sustained by the end consumers while at the same time meeting their needs.

1.4 Research Questions, Thesis Statement and Research Contributions

1.4.1 Research Questions

The main research questions that will be answered in this thesis are as follow:

1. Can the effectiveness of traditional ad hoc retrieval for eDiscovery be enhanced through automatic semantic approach with comparable performance to supervised approaches?

2. Can the vocabulary problem associated with ad hoc retrieval be resolved through semantic approach without hurting efficiency?

Finding answers to the above questions will lead to the following objectives:

1. Topic mining from each of the documents in the collection.
2. Ambiguity resolution of terms in queries and documents.
3. Term expansion of disambiguated terms in the documents with synonyms and related terms.
4. Term expansion of disambiguated queries (derived from production requests) with synonyms and related words.
5. Development of a search and classification system using vector space model of semantics.

1.4.2 Thesis Statement

Ad hoc search in eDiscovery can be conducted using semantic unsupervised technique with competitive effectivess performance with supervised techniques for eDiscovery and improved efficiency performance over state-of-the-art sense-based information retrieval.

1.4.3 Research Contribution

The research contributions are in two fold. Word sense based techniques have been applied to information retrieval (IR) in general to tackle vocabulary problem [32] and not particularly to eDiscovery. Some features distinguishes eDiscovery from other areas of information retrieval among which is that discovery collections being composed mainly of e-mail communications. Furthermore, topic modelling has also been applied to information retrieval but not particularly to eDiscovery.

It is estimated that at least 50% of eDiscovery documents will be in the form of e-mail, with another large chunk coming in the form of office documents (e.g Word, spreadsheets, etc.), together with small databases (e.g MS Access) or larger databases (e.g Oracle), as well as less conventional forms of digitized data (e.g., software code) or other forms (e.g voice mail or video clips) [23]. Topic modelling is one of the best ways to classify topical/conversational discourse like in electronic discovery collections, however, a state-of-the-art topic modelling algorithm; the LDA, though able to detect and group correlated words in documents, cannot specifically resolve the synonymy/polysemy problem in information retrieval (IR). This thesis, therefore provides a novel, gap-filling approach which tackles the synonymy/polysemy issue using word sense disambiguation (WSD) while resolving the computational complexity of word sense disambiguation algorithm through topic modelling for overall efficiency of the approach.

The specific area of contribution of this thesis is the modification of the original Lesk and adapted algorithms for enhanced performance and consequently the application of the modified algorithm in a novel hybrid word sense disambiguation algorithm which combines two distinct but powerful unsupervised algorithms; the modified Lesk algorithm and the Jiang & Conrath similarity measure [46]. The hybrid algorithm outperforms individual constituent algorithms when used in isolation. Furthermore, a novel search and classification algorithm which combines the features of topic modelling and sense ambiguity resolution using the hybrid word sense disambiguation algorithm is proposed and implemented leveraging on the strengths of both. Summarily, the main contribution of this thesis are:

- Development of an independent, state-of-the-art word sense disambiguation (WSD) algorithm through resolution of conflicting sense choice between the modified version of original Lesk algorithm and Jiang and Conrath similarity measure.
- Development of a novel, unsupervised search/classification technique using a hybrid sense and topic based ad hoc information retrieval.

1.5 Organization of the Thesis

The rest of the thesis is organized as follow: Chapter 2 discusses information retrieval (IR), text classification in relation to eDiscovery, query expansion with emphasis on automatic query expansion, approaches and models used in information retrieval/eDiscovery and evaluation in eDiscovery. In chapter 3, we discuss natural language processing (NLP) and its application in eDiscovery, precisely word sense disambiguation (WSD), the original Lesk and adapted algorithms and the Jiang & Conrath similarity measure. We discuss topic modelling and its application in eDiscovery in chapter 4, precisely the latent dirichlet allocation (LDA) model and Gibbs sampling and their applications. The modified and hybrid word sense disambiguation algorithms are discussed in chapter 5. Chapter 6 discusses the combined eDiscovery classification algorithm and the experimental methodology and setting. Empirical evaluation and results of both the hybrid WSD algorithm and the combined unsupervised technique for classification in eDiscovery is the theme of discussion in chapter 7. In chapter 8, we discuss related work from both word sense disambiguation and topic modelling perspectives. Finally, chapter 9 concludes the thesis with recommendations for future work.

Chapter 2

Information Retrieval, Text Classification, Query Expansion and eDiscovery

2.1 Information Retrieval

The field of information retrieval (IR) is a very broad one encompassing other fields of study such as database systems, information extraction, text mining, document filtering among others. Information retrieval (IR) deals with finding materials (usually documents) of an unstructured nature from within large collections (usually stored on computers) that satisfies an information need [66]. Unstructured refers to free natural language text without any defined format which distinguishes it from structured text with defined structure such as the one obtainable in database systems. Text can also be semi structured, which have a partially defined metadata structure such as title, subject, author etc. Information retrieval systems can be categorized according to the level at which they operate. Personal information retrieval can typically take place on a personal computer such as searching for a file using the operating system enabled search. Enterprise search takes place at the level of an organisation such as the one involved in searching an organizations internal documents or employee records and may span over several computers, local network or metropolitan network as the case may be. At the widest extreme is the web search which deals with billions of documents on several millions of computers over the internet. This type of search involves web crawling in gathering these documents from various sources.

2.2 Text Classification

Text classification also known as text categorization or topic spotting is the task of automatically sorting a set of documents into predefined categories. Algorithms or techniques which carry out this task are called classifiers. Given a set of document classes, the task of text classification involves determining which class a particular document belongs to. Examples include automatic detection of an e-mail as either spam or not spam, which topic a news article belongs to such as medicine, business, sports etc. Classification can be done manually by humans, by heuristic rules or based on machine learning. Using heuristic rules typically involves combination of some set of boolean operators and terms to classify text. In supervised machine learning, the set of rules or the decision criteria of the text classifier is learned automatically from training data. In supervised text classification, a number of good example documents (or training documents) is required for each class and the need for manual classification is not eliminated as the training documents come from the person training the system which he does by annotating each document in the training set with a designated class. Supervised text classification algorithms employ feature selection to reduce the high dimensional space of the documents for improved efficiency and scalability. Examples of supervised text classification algorithms include but are not limited to naive bayes classifier (sometimes referred to as semi-supervised), decision trees and support vector machines (SVM).

Unsupervised text classification does not use any labelled example but rather learn from test data itself. Most of the algorithms used for unsupervised text classification are statistical in nature and they include but are not limited to latent semantic indexing (LSI) [25], probability latent semantic indexing (pLSI) [44] and latent dirichlet allocation (LDA) model [11]. In formal terms, given a function $\phi : D \times C \to \{E_1,..,E_n\}$ that describes how documents ought to be classified by means of the classifier ϕ, where $D = \{d_1,.....,d_n\}$ is a set of documents and $C = \{c_1,...,c_n\}$ is a predefined set of categories.

2.3 eDiscovery: In-between Information Retrieval and Text Classifiction

eDiscovery employs techniques used in information retrieval but differs in some features of traditional IR practices and incorporates some text classification features. eDiscovery is a confluence of ideas from information retrieval and text categorization with distinct features from each. Applications of information retrieval techniques in eDiscovery are characterised by the following key challenges [75]:

- eDiscovery emphasizes fixed result sets rather than ranked retrieval applicable in most IR systems such as web search. This is where classification particularly comes in; an electronic document is either relevant or not relevant.
- eDiscovery focuses on high recall even in large collections, in contrast to the high precision focus of many IR applications such as web search.
- eDiscovery evaluation must measure not just relativity, but also absolute effectiveness.

On the other end of the divide is text classification. The challenges of applying unsupervised text classification techniques in eDiscovery include:

- Though the classes are predefined but no additional knowledge is usually available apart from a description of the information need of the user called production request or topic. Unsupervised systems work only with this topic and the document collection while supervised machine learning systems work based on prior manual classification of part of the document collection.
- Unlike labelled text classification where some knowledge of the different classes are used to extract features from the collection to predict a fixed class for documents, eDiscovery employs IR scoring technique and therefore has to determine a cutoff score to place documents in either of the two classes.

2.4 Information Retrieval Models

Mathematical models are used in many scientific areas with the objective to understand and reason about some behaviours or phenomena in the real world. Information retrieval uses these models. Information retrieval models simply define how user queries should be manipulated to meet the information need of the user. There are various retrieval models in the literature, we focus on the three most famous ones and which forms the basis for the model used in this thesis. They include boolean, vector space and probabilistic models.

2.4.1 Boolean Model

Boolean model is based on set theory and boolean algebra. Boolean model expresses queries in the form of logical expression of terms in which the terms are combined using the boolean logical operators "AND", "OR", and "NOT". For instance, to find documents in a collection that contain the words "business" and either "oil" or "gas". We can express it using the booolean model as follows: "business AND (oil OR gas)". Just like in mathematical logic, the product operator "AND" comes before "OR" in the computation order, hence the need to put parenthesis so that the computation priority is given to whatever is in the parenthesis before any other operator. A major advantage of the boolean model lies in its precision, that is, the result is an exact match of whatever is specified in the boolean expression. It is also very simple and very efficient in its implementation since many documents can easily be eliminated using the rules. It is often used in supervised learning systems while hand-crafting some rules that the system needs to follow. Its major drawback is that it does not provide ranking and does not attempt to measure relevance of other documents not directly matched by the boolean expression.

2.4.2 Vector Space Model

Vector space model (VSM) [90] is fundamental to a number of information retrieval applications including document classification, document clustering, document ranking applications such search engines among others. It presents both queries and documents as vectors in a dimensional space. The terms in both are represented along with their frequencies of occurrence and each document is identified by a document identifier such as the file name. In this representation, the vector space is a matrix where the documents each identified by a document identifier are the rows while the columns are the frequencies of the occurrence of terms. The query terms and term frequency of each documents in a collection are expressed separately as a vector in an n-dimensional space where the value of n is the size of the largest document in the collection and the term frequency for non-existent term in a document equals zero. To obtain the similarity score between a query and a document, a similarity computation is done using the document term frequency and computed using cosine similarity. The cosine similarity score is a measure of the angular distance between the query and the document. The angular distance can also be computed between any two documents in the n-dimensional space. Illustratively, the representation of a query and documents both consisting of n-terms in an n-dimensional space is shown in Figure 2.1.

The frequency of a term in a document is called the term frequency, represented as $tf_{t,d}$, where the subscript t is the term and d is the document, uniquely identified by a document identifier. Terms are often presented as bag of words in which the number of occurrence of a particular term and the order is immaterial. That is, each occurring term is treated separately irrespective of whether it appears several times and othe order of appearance does not matter. The inverse document frequency idf is used to scale the score of both frequently and sparsely occuring terms in documents so that one does not have an overbearing effect on the final

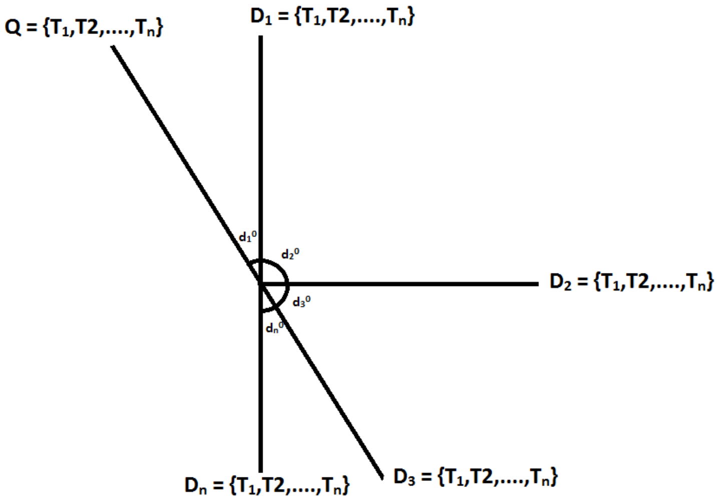
Q = {T$_1$,T2,....,T$_n$}
D$_1$ = {T$_1$,T2,....,T$_n$}
d$_1$^{0}
d$_2$^{0}
D$_2$ = {T$_1$,T2,....,T$_n$}
d$_3$^{0}
d$_n$^{0}
D$_n$ = {T$_1$,T2,....,T$_n$}
D$_3$ = {T$_1$,T2,....,T$_n$}

Fig. 2.1 Vector Space Representation of Query and Documents in an n-dimensional Space.

score. It is calculated using equation 2.1

$$idf_t = log\frac{N}{df_t} \tag{2.1}$$

where N is the total number of documents in the collection and df_t is the number of documents in the collection containing the term t. The weight of each query term t in document d can then be computed using the $tf - idf$ scoring given by $tf - idf_{t,d}$. The $tf - idf$ is then used to assign weight to each term t in a document d using equation 2.2:

$$tf - idf_{t,d} = tf_{t,d} \times idf_t \tag{2.2}$$

The $tf - idf_{t,d}$ assigns weight to a term t, $t\varepsilon Q$ (where Q is the query) in a document d, $d\varepsilon D$ (where D is the document collection) using the following conditions: highest weight when t occurs several times within a few number of documents, lower weight when t occurs fewer times in a document, or occurs in many documents and lowest weight when t occurs in almost all documents.

In the vector view of the query and each document in the collection, the overlap score of a document d in the collection is the summation of the weight of the query terms. Using the $tf - id$f weight, the score is given in equation 2.3:

$$Score(q,d) = \sum_{t\varepsilon q} tf - idf_{t,d} \tag{2.3}$$

2.4.2.1 Scoring with Vector Space Model

The basic idea of scoring with vector space model is derived from the cosine similarity metric. The underlying assumption of cosine similarity for scoring is to normalize the effect of document length on the scores so that the length of any particular document does not lower or increase the score. Cosine similarity uses the dot products of both query and documents and their normalized vectors. The dot product or the inner product is a simple multiplication of constituent query and document vectors.

To compute the similarity score between two documents d_1 and d_2, we compute the cosine similarity between their vectors, which is the angular distance between the two vectors, presented in equation 2.4:

$$sim(d_1,d_2) = \frac{\vec{V}(d_1).\vec{V}(d_2)}{|\vec{V}(d_1)|\,|\vec{V}(d_2)|}, \tag{2.4}$$

where $\vec{V}(d_1).\vec{V}(d_2)$ is the dot product and $|\vec{V}(d_1)|\,|\vec{V}(d_2)|$ is the euclidean lengths of the vectors.The effect of the euclidean length is to normalize the length of the two vectors to unit vectors $\frac{\vec{V}(d_1)}{|\vec{V}(d_1)|}$ and $\frac{\vec{V}(d_2)}{|\vec{V}(d_2)|}$ respectively. Therefore, we have the similarity between documents d_1 and d_2 given in equation 2.5:

$$sim(d_1,d_2) = \vec{d}_1.\vec{d}_2. \tag{2.5}$$

To compute similarity between the query vector and any document in the collection which is the similarity score for that document, we use equation 2.6.

$$sim(q,d) = \vec{q}.\vec{d} \implies score(q,d) = \frac{\vec{V}(q).\vec{V}(d)}{|\vec{V}(q)\,|\vec{V}(d)|}. \tag{2.6}$$

2.4.3 Probabilistic Models

Probabilistic models are models that evaluate the likelihood of a document meeting the information need of a user by using probability theory to compute their similarity with the query. These models include binary independence model, okapi BM25 weighting model [49], bayesian network models etc. Some are based on natural language and they include unigram, bigram and ngram models. One of the most popular of these models which is based on spatial representation include the query likelihood model and the Kullback-Leibler model [54]. They also include probabilistic topic models such as probabilistic latent semantic indexing (pLSI) and latent dirichlet allocation (LDA) model.

2.5 Query Expansion (QE)

Query expansion is one of the standard methods for improving performance in text retrieval applications. A number of the highly ranked documents from the original query are reissued as new query. In this way, additional relevant terms can be added to the initial query. Query expansion refers to the technique that uses blind relevance feedback to expand a query with new query terms, and reweigh the query terms, by taking into account a pseudo relevance set [86, 3]. The pseudo relevance set consists of the top ranked documents returned by the first-pass retrieval. The top-ranked documents are assumed to be relevant to the topic. Query expansion has proved to be an effective technique for ad-hoc retrieval. It is an effective approach to alleviate the vocabulary problem [32] in ad hoc retrieval. It involves the extraction of useful terms from the top retrieved documents. This is sometimes referred to as retrieval feedback or pseudo relevance feedback. It is generally believed that query expansion can greatly enhance the recall of systems but past experiments has shown that if properly managed can also maintain or improve precision. Past experiments reported in the literature, reviewed by [41, 89] have shown that the application of this technique often results in a loss in precision higher than the corresponding gain in recall. However, some successes have been reported in applications to large-scale collections [12, 103, 108, 109]. In fact, almost all the groups involved in the TREC evaluations over the years have reported improved results by expanding queries using information from the top retrieved documents [100, 102]. Some reported improvements in the order of 10% and above [62, 71, 17, 57]. The growing interest in this technique makes evident the need to develop well-founded methodologies for ranking and weighting expansion terms and to perform experimental studies for evaluating and contrasting merits and drawbacks of currently used methods for query expansion, rather than just making comparisons with the use of non expanded queries.

2.5.1 Automatic Query Expansion (AQE)

A major drawback of relevance feedback is that performance of systems may be hurt if top ranked documents have low precision. This is so because most top ranked documents may not actually be relevant since the first pass of relevance feedback works based on keyword search which is often affected by vocabulary problem, hence there is need for automatic query expansion which enriches the initial query with semantically related terms without having to manually or interactively search through the first pass documents.

To deal with this vocabulary problem, several approaches have been proposed including interactive query refinement, relevance feedback, word sense disambiguation, and search results clustering [16]. However, for improved precision on the recall success of query expansion, there is need to distinguish among different senses of words and their meanings. This is exactly what WordNet [70, 31] provides which serves as the knowledge resource for the hybrid disambiguator on which this work is partly based. Word sense disambiguation (see section 3.2) helps to identify the meaning of each query and document term so that they can be correctly expanded with related terms. This helps in improving both recall and precision. This thesis presents an automatic query expansion from initial query using a hybrid word sense disambiguation algorithm.

2.5.2 Related Techniques to Automatic Query Expansion

In the context of automatic query expansion, [16] identified alternative strategies to curb the vocabulary problem include the following:

1. **Interactive Query Expansion and Refinement**
   Interactive query expansion (IQE) and refinement as presented in [29, 7] differs from automatic query expansion (AQE) in that the various alternatives for query (re)formulation are decided by the user. Interactice query expansion has the potential for producing better results than automatic query expansion [53], but this generally requires expertise on the part of the user [87]. It offers more user control over query processing. However, it is very costly in terms of time, speed and effort.

2. **Relevance Feedback (RF)**
   Relevance feedback takes the results that are returned from a first pass retrieval and uses relevance information provided by the user to reformulate the initial query in manner that enhances performance. The content of the assessed documents is used to adjust terms' weights and/or add related and similar words to the initial query. The major difference between relevance feedback and automatic query expansion is that relevance feedback tries to expand initial query with related and similar words based on user response while automatic query expansion tries to do same without the user input; it automatically determines the query expansion words.

3. **Word Sense Disambiguation in Information Retrieval**
   Word sense disambiguation (WSD) tries to be specific in reformulating and/or expanding initial query by determining the sense of query terms. (See sections 3.2 and 8.1 for full discussion on word sense disambiguation and its various application approaches in information retrieval (IR)).

4. **Search Results Clustering (SRC)**
   Search results clustering arranges search results by topics thereby allowing access

to specific aspects of the query. In contrast to traditional clustering, search result clustering techniques focus on both clustering structure and the quality of the cluster labels. Kurland et al. [53] makes a specific link between search result clustering and automatic query expansion using built clusters from top ranked documents in search results as pseudo-queries representing distinct feature of the original query.

2.6 eDiscovery Models

eDiscovery models define the standards, stages and processes and evaluation metrics involved in conducting eDiscovery business from the request for production to the actual production of the relevant documents. There are several models used in eDiscovery but we discuss the three relevant to this thesis. They include the EDRM metrics model, the EDRM reference model and the eDiscovery pyramid.

2.6.1 The Metrics Model

The EDRM, a coalition of consumers and providers working together since 2005 to create practical resources to improve eDiscovery and information governance proposed the metrics model. The EDRM metric model [28] provides a framework for planning, preparation, performance and follow-up of eDiscovery issues and projects by depicting the relationship between the eDiscovery process and how information, activities and outcomes may be evaluated. The model is presented in Figure 2.2.

The Metrics Model consists of two inter-dependent elements: the centre, which includes the key metrics variables of volume, time and cost, and the elements at the outside nodes, which indicate seven aspects of work which affect the outcome associated with the elements at the centre. The seven work aspects are:

1. **Activities**: These are things that are being done by either people or technology. Examples include: documents collection, designing a search, interviewing a custodian, etc. Monitoring activities is essential to ensure timely and cost-efficient project completion. Pre-planning activities provides a basis for budgeting, understanding, and allocating the resources, time and money required for the eDiscovery project.

2. **Custodians**: Custodians are people having administrative control of documents or electronic files or systems; for example, the custodian of an email is the owner of the mailbox which contains the message. The number of custodians can be a significant factor in the volume of data to be collected and can therefore affect costs and may also

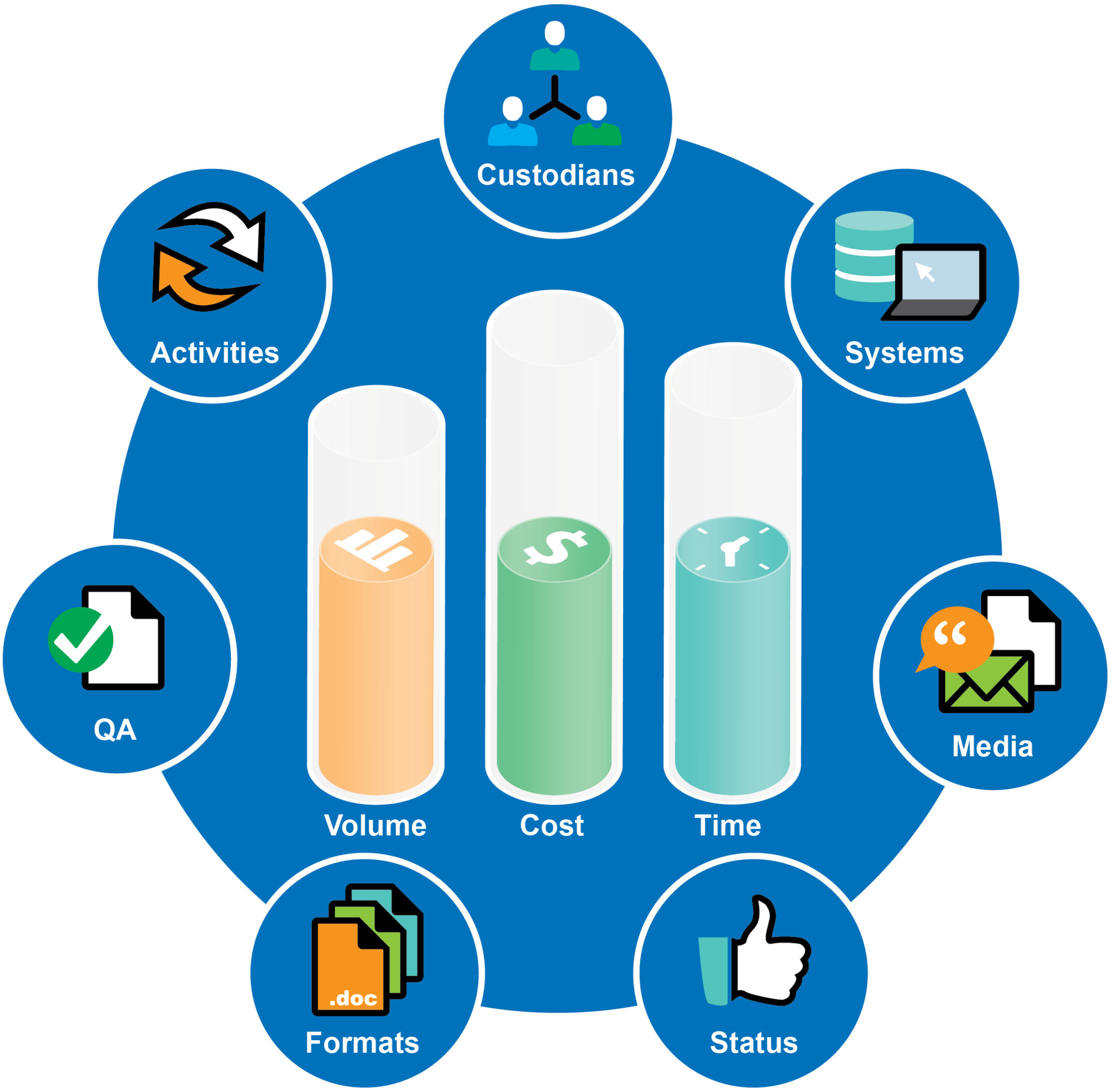

Fig. 2.2 The Metrics Model.

affect the time required to conduct interviews and collections and may increase the complexity and costs associated with preserving and managing data.

3. **Systems**: These are the places, technologies, tools and locations in which electronic information is created, stored or managed; examples of systems include shared drives, email, computer applications, databases, cloud sources and archival sources such as back-up tapes. The number of systems implicated in a preservation or collection effort can contribute to larger volumes of ESI and increase costs, processing time or complexity and affect the time required and cost of subsequent steps.

4. **Media**: These are the storage devices for electronic information; examples include: CDs, DVDs, floppy disks, hard drives, tapes and paper. In order to ensure that all the relevant information are accounted for, media tracking is utilized. Furthermore, data volumes can affect the choice of media.

5. **Status**: A unique point in time in a project or process that relates to the performance or completion of the project or process; measured qualitatively in reference to a desired outcome. Regular tracking of projects against desired milestones can contribute to greater efficiency, time and cost saving measures. Ensuring that a project is on track can facilitate better budgeting and planning. It can also contribute to effective communication with all parties involved, including the court.

6. **Formats**: This refers to the way information is arranged or set out; for example, the format of a file which affects which applications are required to view, process, and store it. The format in which documents originate, and move through the EDRM project lifecycle can be a factor which influences the costs and the time required to complete tasks. Decisions need to be taken regarding formats of data, media, production and presentation, bearing in mind the impact of the choice on task or project completion.

7. **Quality Assurance (QA)**: Ongoing methods to ensure that reasonable results are being achieved; an example of QA would be to ensure that no privileged documents are released in a production by performing an operation, such as checking for privilege tags within the production set. Quality Assurance can encompass many activities, but refers to those that are performed to control the quality or outcome of a task or project. Quality Assurance is a recommended best practice in all phases of the EDRM lifecycle. Implementing quality assurance in projects can reduce cost by ensuring that the work is completed timely and accurately, and can contribute to lowering the time required to complete tasks and projects.

2.6.2 The EDRM Electronic Discovery Reference Model

The electronic discovery reference model [27], also a proposition by the EDRM group, presents a diagramatic illustration of a conceptual view of the eDiscovery process. It is presented in Figure 2.3.

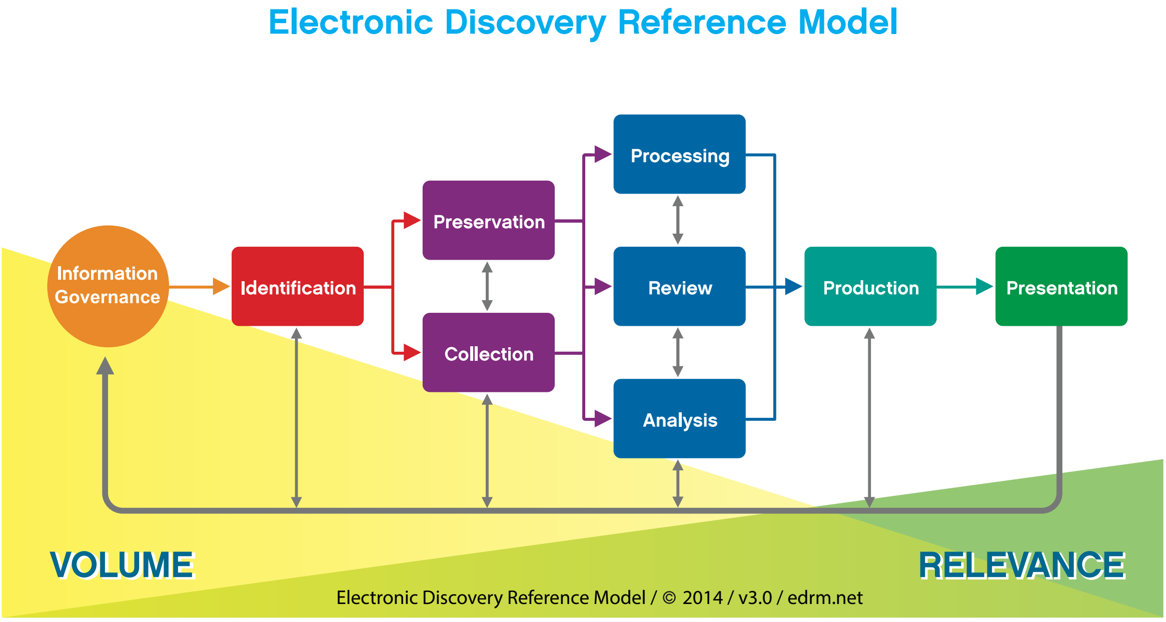

Fig. 2.3 The Electronic Discovery Reference Model.

In practice, each of the steps might be repeated a number of times. As the authors put it, it was developed for discussion and analysis purposes and not a prescription of a monopolistic way of conducting eDiscovery.

1. **Information governance**: The first step is the information governance, which involves management of electronic collections in preparation of probable eDiscovery matters.
2. **Identification**: Identification is the location of potential sources of ESI and determining its scope, breadth and depth.
3. **Preservation**: Preservation ensures protection of ESI against alteration and/or destruction.
4. **Collection**: Involves gathering of ESI for use in eDiscovery.
5. **Processing**: This is the task of converting ESI from one form to another in preparation for classification.

6. **Review**: Checking ESI for relevance or privilege.
7. **Analysis**: Evaluating ESI for content and context, including key patterns, topics, people and discussion.
8. **Production**: Delivering ESI to the requesting party in the appropriate forms.
9. **Presentation**: Displaying ESI before audiences in hearings, trials, etc., especially in native and near-native forms, to elicit further information, validate existing facts or positions, or persuade an audience.

Each of the steps as defined by the model may involve one or more substeps and some tasks in the discovery process may be a combination of two or more steps. For instance, the actual classification of ESI may combine processing and review.

2.6.3 The eDiscovery Pyramid

The eDiscovery pyramid [21] summarizes the processes involved in eDiscovery. It presents a precise insight and focus more on the search, classification and production of documents. It is categorized into four structures as presented Figure 2.4:

1. **Foundation Tier (Hosting)** — Collecting: including identification, conversion and migration.
2. **Second Tier (Indexing)** — Vetting: including filtering, deduplication, managing similar objects.
3. **Third Tier (Searching and Navigating)** — Organizing: including classifying and clustering; tagging and linking related documents.
4. **Fourth Tier (Reporting)** — Analyzing: production; including consolidating and summarizing findings.

The activities in each tier are enclosed in parenthesis while the techniques used for them are explain afterwards. The benefits that arise from viewing eDiscovery like this pyramid include being able to build creatively upon the foundations established in the preceding tasks. This alternative model may be more suitable for researchers who are attempting to tackle difficult precision and recall problems and reporting requirements, in contrast to engineers who are expected to satisfy operational constraints while moving the pipeline closer to production [21].

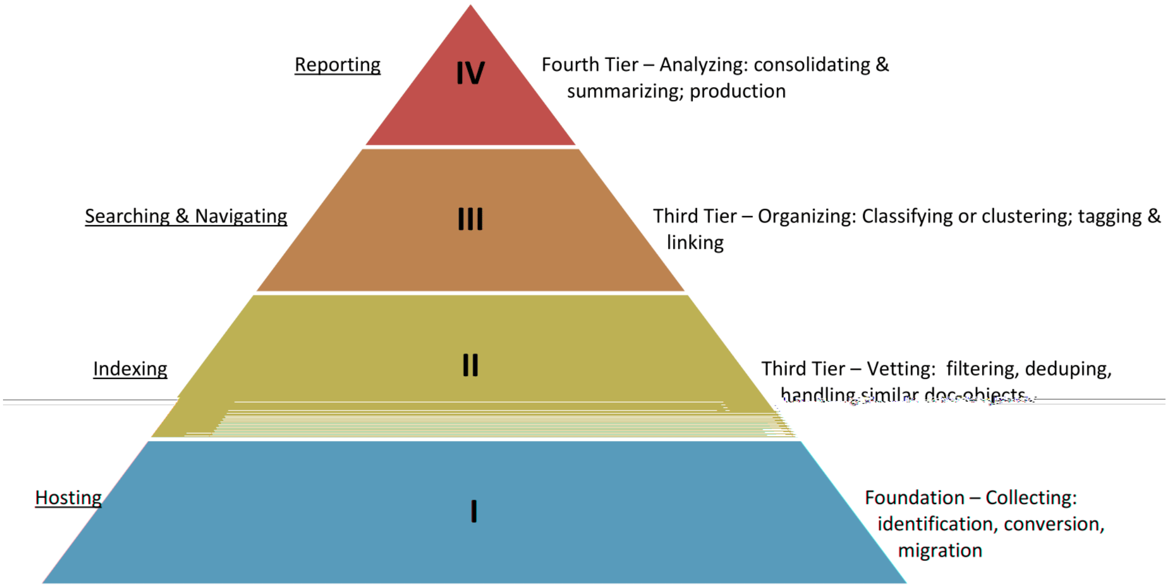
Reporting
IV
Fourth Tier – Analyzing: consolidating & summarizing; production
Searching & Navigating
III
Third Tier – Organizing: Classifying or clustering; tagging & linking
Indexing
II
Third Tier – Vetting: filtering, deduping, handling similar doc-objects
Hosting
I
Foundation – Collecting: identification, conversion, migration

Fig. 2.4 The eDiscovery Pyramid - A Technology Perspective.

2.7 Evaluation in eDiscovery

Evaluation for effectiveness in eDiscovery is basically the one borrowed from Information Retrieval. The evaluation metrics measure the degree to which the documents retrieved as relevant by eDiscovery search systems actually satisifies the information need of the requesting party.

Cleverdon and Keen [20] identified five key factors that affects an IR systems when a user submits a query to it to the time which the system outputs a ranked list of documents in response to the query. These factors include:

1. "The ability of the system to present all relevant documents"
2. "The ability of the system to withhold non-relevant documents"
3. "The interval between the demand being made and the answer being given (i.e., time)"
4. "The physical form of the output (i.e., presentation)"
5. "The effort, intellectual or physical, demanded of the user (i.e., effort)"

Sanderson [92] added to the list of these factors as follows:

- "The ability of the user at specifying their need"
- "The interplay of the components of which the search algorithm is composed"
- "The type of user information need"
- "The number of relevant documents in the collection being searched"
- "The types of documents in the collection"
- "the context in which the user's query was issued; and"
- "the eventual use for the information being sought."

Evaluation metrics for information retrieval/ eDiscovery systems can be categorized into three depending the purpose and goal of the system; effectiveness measures, efficiency measures and datasets.

2.7.1 Effectiveness

The effectiveness of an IR system centres on how accurately the system is able to retrieve relevant documents that meet the information need of the user. Relevance is key to the effectiveness of any IR system which means how best the documents retrieved answers the information need of the user. Information retrieval relies on datasets of documents whose relevance for a given query have been adjudged by human(s). Unfortunately, there is no universal definition of what a relevant document is: the notion of relevance is divergent because the same document can have different meanings to different humans. This has been discussed by [101] who noticed discrepancies between relevance judgments made by different annotators. However, post-adjudication agreement of divergent views of annotators are often carried out to arrive at the final relevance judgement, taking the level of inter-annotator agreements into consideration.

2.7.1.1 Types of Relevance

Magalhães [65] identifies three types of relevance for evaluation of multimedia information retrieval which is also applicable to information retrieval in general. They include:

- **Binary relevance**: In binary relevance, a document is either relevant or not. It makes the simple assumption that relevant documents contain the same amount of information value. This is exactly the case in the determination of relevance of documents in eDiscovery. The category a documents belongs is, however determined by a specified cutoff point.

- **Multi-level relevance**: Multi-level relevance specifies that documents contain information with different degree importance for the same query. Therefore, a discrete model of relevance (e.g. highly relevant, relevant, moderately relevant, not relevant) enables systems to rank documents by their relative importance. This type of relevance judgment allows annotators to rate documents with different levels of relevance for a particular query.

- **Ranked relevance**: This is similar to multi-level relevance but ranks documents with the assumption of ordering documents from the most relevant to the least relevant. It does not use predetermined number of classes as aplicable in multi-level relevance as the number of ranked documents vary from query to query.

2.7.1.2 Incomplete and Inconsistent Relevance Judgements

A major practical problem to relevance determination is incompleteness or inconsistency in relevance judgements [65]. Incomplete relevance judgement often arise as a result of very large collections, where human assessors cannot judge all possible documents in the collection. Inconsistency in relevance judgement also arises as a result of disagreement among the assessors on what is relevant and what is not. Voorhees [101, 13] proposed a metric to reduce the effect of incomplete relevance judgments. Aslam and Yilmaz [6, 110] proposed more stable metrics to tackle the stability of measures under incomplete and inconsistent relevance judgments.

However, studies have shown that assessors' inconsistencies does not have much impact on the document ranking for relevance. Lesk and Salton [88] studied the issue of assessor consistency by gathering a set of relevance judgements for a test collection comprising 1,268 abstracts and 48 queries. In their study, they compared and contrasted the judgements of the creators of the queries with the judgements of "subject experts" in the field who independently assessed the documents for relevance. They experimented with three configurations of a search engine on the pair judgement set and concluded that regardless of the configuration used, results ranking remain same. Their reason for this consistency was because assessors's relevance disparity are always found at the least ranked documents.

Cleverdon [19] used relevance judgements on Cranfield II collection, and also concluded that annotators' assessment disparity did not have much effect on search results ranking of different configuration of a retrieval system.

2.7.2 Efficiency

IR systems dealing with large collection are often faced with scalability and computational complexity issues. Complexity can arise in any phase of the entire search process such as document analysis, query analysis among others. Usually efficiency of a system is measured in terms of response time, which is the time between submission of a query and production of results and the memory space needed to process search. In the context of this work, the main efficiency issue is on the efficiency of the word sense disambiguation algorithm. The time taken for running the hybrid WSD algorithm on the representative test collection is 5 weeks when duplicated into 10 runs, which runs simultaneously. Following Magalhães [65], we focus on the documents indexing and query analysis complexities.

2.7.2.1 Indexing Complexity

Indexing is the processing of the original data into a highly efficient cross-reference lookup in order to facilitate rapid searching. Indexing makes the whole search time efficient, in contrast to serial consideration of documents in a collection for search.

2.7.2.2 Query Analysis Complexity

The query analysis complexity refers to the cost of processing the standard query-parsing methods in traditional IR systems. The IR model being implemented determines how the query will be parsed.

2.7.3 Collections

Collections are research tools that provide a common test environment to evaluate and compare the effectiveness and efficiency of different algorithms. Collections may be in the form of texts (e.g. Reuters corpus version 1 [60], 20 news group [55], enron dataset), images (e.g. Caltech 256), music and videos (e.g. movie review collection [78]). In eDiscovery, for instance the Enron dataset is considered to appropriately model the real-life electronic discovery collection scenario since the bulk of eDiscovery documents are in the form of e-mail including their attachments and large enough for experimental purpose. It has been the test collection for the latest editions of TREC legal track. The effectiveness performance of an IR system may be dependent on the test collection. This is because the performance may not be consistent for different test collections.

Sanderson [92] itemizes the classic components of test collections as follows:

- A set of documents, each uniquely identified by an identifier denoted *docid*.

- A set of topics, also called queries, each uniquely idenfied by an id, denoted *qid*.
- A set of relevance judgements, denoted *qrels* with details of documents relevant to a particular topic, usually comprising of *qid* and *docid* pairs.

Test collections and evaluation metrics were developed and tailored to reflect the changing priorities and needs of IR researchers and consequently, works in the two areas are described in chronological order [92] as follows:

- **Early 1950s–early1990s**: This era witnesses the initial development of test collections and evaluation metrics. Collections of this time are composed mainly of catalogue information scholarly publications and news articles and the evaluation focus is on high recall.
- **Early 1990s–early 2000s**: This era witnesses the development of large collections by the IR research community and the evaluation focus is still on high recall.
- **Early 2000s–present**: This era witnesses the development of more and larger collections for evaluation purposes. The evaluation focus is the retrieval of relevant documents as small as the total number of relevant documents in a collection. This leads to the F-measure (F1) being used to balance the degree of recall and precision.

2.7.3.1 Challenges of Building and Using Information Retrieval Test Collections

Sanderson [92] identifies the following challenges faced when developing a test collection:

1. **Problem in Obtaining Documents**: Gathering digital documents from various sources in different formats is a very difficult task to accomplish. Some documents are even required to be converted from their hardcopy forms to digital forms. An example is the IIT CDIP Test Collection [59], used for earlier editions of TREC legal track, where hardcopy documents needs to be scanned and preserved as disk images for usage in IR evaluation. Generally, original documents either digital or hardcopy, need one form of processing or the other while developing them into a test collection .
2. **Distribution of Collections**: Researchers want to share test collections for evaluation of their works, however, there exist some challenges in the past (example is early TREC tasks), where test collections are stored in storage disks and distributed to only participants. This limits effective distribution where non-participants who need it in future find it difficult to obtain. Also, they are easily prone to damage. Recent evaluation events now have dedicated sites where test collections can be downloaded

though some are later left unmaintained after a period of time which also hinders easy access. Another problem is that some collections' developers requires financial compensaton for release of test collections which also limits easy distribution.

3. **The Problem of Scale**: Early IR systems use keywords search over collections typically tens of thousands of documents in magnitude. By mid 1970s, the magnitude has increased to hundreds of thousands of documents. A major reason for lack of unwillingness in building and using large test collections is the insistence on knowing the actual number of relevant documents which becomes difficult with the increasing size of collections. There is , however, a convergence of agreement among IR researchers that solutions must be found to develop larger test collections at the same time identify as many possible relevant document for queries on the collections. Spärck-Jones and Van Rijsbergen [48], suggest a methodology for developing larger test collections, which they call *ideal test collection*. There proposition was propelled by concern that existing test collections were not only small but often incautiously built and/or inappropriately used by researchers. They proposed the creation of more large test collections built using tenable principles and distributed to the community by a common organization. In addressing the problem of assessors not being able to judge every document in large collections or disparity in their judgements, they proposed a solution using a method they referred to as pooling. In pooling, a small subset of documents containing a sufficiently representative sample of the relevant documents would be created. In another work, Spärck-Jones and Bates [47] later investigated and analyzed the impact of pooling on some information retrieval systems.

A major challenge we identify in the usage of test collections is detailed below:

- **Inappropriate Documents Formats For Information Retrieval Task**: With particular reference to the IIT CDIP Collection [59] used for earlier editions of TREC legal track up till tear 2008 task,

2.7.3.2 Main eDiscovery Effectiveness Evaluation Metrics

In practice, it is difficult to determine effectiveness of IR systems because no one has an idea of the actual number of relevant documents in a collection, however, just like in other fields in IR, tasks are usually organized to simulate evaluation where a collection of a known size is manually searched and analyzed for relevant documents. From this analysis, the effectiveness of an algorithm or a system can be measured by calculating the number of relevant documents it retrieves measured using the manual human judgement. The main evaluation metrics used in eDiscovery are precision, recall and F-measure (F1).

- True Positive: This is the number of relevant documents in the collection correctly classified as relevant, denoted TP.

- True Negative: This is the number of irrelevant documents in the collection correctly classified as irrelevant, denoted TN.

- False Positive: This is the number of irrelevant documents in the collection incorrectly classified as relevant, denoted FP.

- False Negative: This is the number of relevant documents in the collection incorrectly classified as irrelevant, denoted FN.

1. **Precision**

   Precision is the proportion of the actually relevant documents out of the total number retrieved as relevant by the system given by equation 2.7.

   $$Precision = \frac{\text{number of relevant documents retrieved}}{\text{number of retrieved documents}} \tag{2.7}$$

   Using the parameters we defined earlier, this results in equation 2.8.

   $$Precision = \frac{TP}{TP+FP} \tag{2.8}$$

2. **Recall**

   Recall is the proportion of the actually relevant documents retrieved by the system out of the total number of relevant documents in the collection calculated using equation 2.9.

   $$Recall = \frac{\text{number of relevant documents retrieved}}{\text{number of relevant documents in the collection}} \tag{2.9}$$

   Using the parameters we defined earlier, this results in equation 2.10.

   $$Recall = \frac{TP}{TP+FN} \tag{2.10}$$

3. **F1**

   F1 is a metric which balances the effect of precision and recall. Depending on the IR application, either of precision or recall is preferred to the other, however, when two systems differ in their results for both precision and recall, F1 is used to judge which is better considering the effect of the two metrics. For instance, a system which returns the whole documents in a collection because recall is preferred will have 100% recall with 0% precision or a system which returns a single document out of the 1000 relevant documents in a collection because precision is preferred will have 100% precision with 0.001 recall. These two illustrations does not practically make sense. Ideally, precision decreases as the number of documents retrieved is increased while recall increases. To neutralize the effect of this anomally, the F1 metric balances the tradeoff between the two metrics. F1 is calculated by equation 2.11.

$$F1 = \frac{2}{\frac{1}{R} + \frac{1}{P}} \tag{2.11}$$

where P and R are precision and recall respectively

From the above, we derive equation 2.12.

$$Recall = \frac{2PR}{P+R} \tag{2.12}$$

Considering our examples above, using the F1 metric, each of the two cases will result in F1 of 0% which depicts the senselessness in the two scenarios.

In any retrieval system there is always a tradeoff between precision and recall because of the difficulty in obtaining a balanced effect of the two metrics. eDiscovery favours recall over precision. We aim to develop a technique applicable to all cases irrespective of the dataset or collection involved with improved recall while maintaining a reasonable degree of precision to balance the recall success.

2.7.3.3 Other Effectiveness Evaluation Metrics

1. **Elusion**

   Elusion in eDiscovery simply means the proportion of relevant unretrieved document in the collection. It is calculated using equation 2.13.

$$Recall = \frac{FN}{FN+TN} \tag{2.13}$$

   Or simply $1 - R$, where R is the recall.

Other metrics such as Receiver Operating Curve (ROC), Area Under Curve (AUC) are of no particular importance to eDiscovery.

Chapter 3

Natural Language Processing & eDiscovery

3.1 Natural Language Processing (NLP)

Natural language processing (NLP) is a branch of Computer Science, Artificial intelligence and Computational linguistics which deals with developing systems that allow computers to understand and manipulate human language. Because of the human language, hence the use of the word *"natural language"* which distinguishes it from artificial languages such as programming languages. There are vast amount of resources in several natural language these days for which the computer can be used to process for human use. There are several research activities aimed at making computers understand and use available information and resources to do specific tasks for humans. There are three major aspects of natural languages: the syntax, the semantics and the pragmatics. The syntax deals with language form usually specified by a grammar. Semantics deals with the meaning aspect of the language while the pragmatics describes the spoken aspect of the language and how it relates to the world. NLP application areas include Information Retrieval, Text Classification, Information Extraction, Text Summarization, Speech Recognition, Word Sense Disambiguation, Dialogue Systems, Question Answering among numerous others. In the next section, we discuss Word Sense Disambiguation (WSD), a major underlying component of the techniques used in this thesis.

3.2 Word Sense Disambiguation

Ambiguity is a fundamental characteristic of almost every language of which English Language is not an exception. A considerable number of English words have more than one meaning and quite a number share the same spelling but differ in meaning. The context of usage of these words defines the meaning intended by the user. Word sense disambiguation (WSD) is an AI-hard problem and is considered to be an intermediate step in many NLP tasks. Speakers and writers of natural languages do not have difficulties in inferring the meaning of words used in any context but this is difficult for machines which have to process a lot of data, learn from examples or knowledge resources and implement complex algorithms in order to determine the meanings of ambiguous words used in natural language expressions. However, we need these machines to do a lot of work for us as humans due to the ability of computers to do work that are difficult for humans to do manually such as searching a document collection consisting of several millions of electronic documents, tasks involving complex computations and/or repetitive in nature etc.

The computational identification of the meaning of words used in different contexts is called word sense disambiguation (WSD), also known as Lexical Disambiguation. We consider the following sentences as instances of the usage of the word *"bank"*:

1. *My bank account yields a lot of interest annually.*
2. *The children are playing on the bank of the river.*

Based on the context of usage of the word *“bank”* in the two sentences above, we can infer that the first instance i.e sentence 1. refers to a financial institution where money is kept and financial transactions are done while the second instance, sentence 2. refers to a sloppy land beside a river. Human identification of the correct sense of the word *“bank”* as used in the two sentences is relatively easy for people proficient in English, but not so with machines which need to process large unstructured textual information, using knowledge from training documents or lexicons, carrying out complex algorithmic computations in order to determine the correct sense of this word as used in the two statements. Basically, the output of any word sense disambiguation system is a set of words sense-tagged with the right synonymous word (if any). Considering the instances in the examples above, the sentences can be sense-tagged as follow:

1. *My bank/financial institution/banking concern account yields a lot of interest annually.*
2. *The children are playing on the bank/sloppy land of the river.*

Word sense disambiguation relies on knowledge. This means it uses a knowledge resource or sources to associate the most appropriate senses with the words whose sense determination is under consideration. Ideally, Word sense disambiguation is a means to an end but not usually the end itself, enhancing other tasks in different fields and application development such as parsing, semantic interpretation, machine translation, information retrieval and extraction, text mining, and lexical knowledge acquisition. Word sense disambiguation has been applied extensively in information retrieval but not particularly in eDiscovery which has some distinct features discussed earlier in Chapter 2.

Approaches to word sense disambiguation may be supervised or unsupervised. Supervised approaches use machine learning techniques to develop a system from labelled training sets which are then applied to a test dataset. One disadvantage of supervised approach is that the performance of systems tend to degrade when they encounter unseen words in test dataset or when a completely new test set is used for evaluation. Unsupervised approaches on the other hand, use unlabelled corpora in system development, and do not exploit any manually sense-tagged corpus to provide a sense choice for a words in any usage context.

Another approach is Knowledge-based approach which is a form of unsupervised approach. Knowledge-based approaches depend on some knowledge dictionary or lexicon to computationally identify the meaning of the words used in a particular context. While knowledge-based and unsupervised approaches are automatic and do not involve manual training of systems, supervised approaches involve the herculean task of manually training systems with labelled examples. Supervised methods also suffer from sparseness in training data, in contrast to knowledge-based methods, which use any of the many available resources of tagged data.

In this thesis, we employ a hybrid knowledge-based approach which heuristically resolves any conflict between the two constituent algorithms making the hybrid algorithm. In the following sections, we discuss the constituent and the hybrid algorithms.

3.2.1 The Original Lesk Algorithm

Michael Lesk invented this approach named gloss overlap or the Lesk algorithm [58]. It is one of the first algorithms developed for the semantic disambiguation of all words in unrestricted texts. The only resource required by the algorithm is a set of dictionary entries, one for each possible word sense, and knowledge about the immediate context where the sense disambiguation is performed. The idea behind the Lesk algorithm represents the seed for today's corpus-based algorithms. Almost every supervised WSD system relies one way or the other on some form of contextual overlap, with the overlap being typically measured between the context of an ambiguous word and contexts specific to various meanings of that word, as learned from previously annotated data provided by the knowledge resources being employed.

The main idea behind the original definition of the algorithm is to disambiguate words by finding the overlap among their sense definitions. Namely, given two words, W_1 and W_2, each with NW_1 and NW_2 senses defined in a dictionary, for each possible sense pair W_{1_i} and W_{2_j}, $i = 1,......,NW_1$, $j = 1,......,NW_2$, we first determine the overlap of the corresponding definitions by counting the number of words they have in common. Next, the sense pair with the maximum overlap is selected, and therefore the sense is assigned to each word in the text as the appropriate sense. Several variations of the algorithm have been proposed after the initial work of Lesk. The original Lesk algorithm is summarized below:

```
1. for each sense i of $W_1$
2.     for each sense j of $W_2$
3.         compute overlap(i,j), between the definitions of sense i and sense j
4.         find i and j for which overlap(i,j) is maximized
5. assign sense i to $W_1$ and sense j to $W_2$
```

Our modification of this algorithm followed the work of Banerjee and Pedersen [8] who adapted the algorithm using WordNet [70, 31] and the semantic relations in it.

3.2.1.1 Work Based on Original Lesk Algorithm

Since the seminal work of Michael Lesk [58], several variations of the algorithm have been implemented. Each variation of the algorithm either (1) attempts to solve the combinatorial explosion of possible word sense combinations when more than two words are considered or (2) attempts to disambiguate, where each word in a given context is disambiguated individually by measuring the overlap between its corresponding dictionary definitions and the current sentential context and alternatives where the semantic space of a word meaning is augmented with definitions of semantically related words.

Cowie et al. [24] worked on a variation called simulated Annealing. In this work, they define a function E that reflects the combination of word senses in a given text whose minimum should correspond to the correct choice of word senses. For a given combination of senses, all corresponding definitions from a dictionary are collected, and each word appearing at least once in these definitions receives a score equal to its number of occurrences. Adding all these scores together gives the redundancy of the text. The E function is then defined as the inverse of redundancy. The goal is to find a combination of senses that minimizes this function. To this end, an initial combination of senses is determined (e.g. pick the most frequent sense for each word), and then several iterations are performed, where the sense of a random word in the text is replaced with a different sense, and the new selection is considered as correct only if it reduces the value of the E function. The iterations stop when there is no change in the configuration of senses.

Banerjee and Pedersen developed a variation of the Lesk algorithm, called the Adapted Lesk algorithm [8] which uses WordNet [70, 31] as the knowledge resource. Part of our technique builds on this, so we elaborate more on it than on other variations. Definitions of semantically related word senses in WordNet lexical hierarchy are used in addition to the definitions of the word senses themselves to determine the most likely sense for a word in a given context. Banerjee and Pedersen employed a function similar to the one defined by Cowie et al. [24] to determine a score for each possible combination of senses in a text and attempt to identify the sense configuration that leads to the highest score. While the original Lesk algorithm considers strictly the definition of a word meaning as a source of contextual information for a given sense, Banerjee and Pedersen extended this algorithm using WordNet [70, 31]. In their work, they employed WordNet synsets to obtain word senses and their meanings (through their glosses). In addition, related concepts and the definitions of each word sense based on semantic relations in WordNet were also used. These relations include hypernymy, hyponymy, meronymy etc. In their work, a limited window size of context words was used by considering only the immediate words before and after the target word. The algorithm takes as input an example or an instance in which target word occurs, and it will produce the sense for the word based on information about it and few immediately surrounding words. The crux of the work is the adaptation of WordNet semantic relations to the original Lesk algorithm and computation of similarity based on maximum overlap of words (taking into account the number of times each word appears in the glosses).

In this work, we have employed a unlimited window size for words in context, that is, all the surrounding words in a sentence are considered and consequently used in computing overall cumulative similarity for words senses under consideration while adding useful WordNet relations. (See section 5.1)

3.2.1.2 WordNet

WordNet is a manually-constructed lexical database system developed by George Miller [70, 31] and his colleagues at the Cognitive Science Laboratory at Princeton University and also made available in electronic form. The basic object in WordNet is a set of synonyms called a synset. By definition, each synset in which a word appears is a different sense of that word. There are four main divisions in WordNet, one each for nouns, verbs, adjectives and adverbs. Within a division, synsets are organized by the lexical relations defined on them. For nouns, the lexical relations include antonymy, hypernymy/hyponymy (IS-A relation) and three different meronymy/holonymy (PART-OF) relations. The verb also include the antonymy, hypernymy/hyponymy, troponymy and other relations like entailment, causes etc. The IS-A relation is the dominant relation, and organizes the synsets into a set of hierarchies. WordNet can be said to be the standard knowledge source for word sense disambiguation in English. This is as a result of several factors which include the fact that is not domain specific, very extensive and publicly available. For instance, the word "game" has the following noun synsets: "game.n.1", "game.n.2", "game.n.3", "game.n.4", "game.n.5", "game.n.6", "game.n.7", "game.n.8", "game.n.9", "game.n.10", "game.n.11". For each sense in synset, WordNet has the following information, some depending on the part of speech:

1. **Gloss**: This is a textual definition of each sense in the synset possibly with a set of usage examples

2. **Antonymy**: This is a semantic relation which exist between senses. X is an antonym of Y if it expresses the opposite concept. Antonymy applies to all parts of speech.

3. **Pertainymy**: X is an adjective which can be defined as “of or pertaining to” a noun (or, occasionally, another adjective).

4. **Hypernymy**: Y is a hypernym of X if every X "is a kind of" Y. Hypernymy holds between pairs of noun or verb synsets.

5. **Hyponymy and troponymy**: This is the inverse relations of hypernymy for noun and verb synsets, respectively.

6. **Meronymy**: Y is a meronym of X if Y "is a part of" X. Meronymy holds for noun synsets only.

7. **Holonymy**: Y is a holonym of X if X "is a part of" Y. That is, the inverse of meronymy.

8. **Entailment**: A verb Y is entailed by a verb X if by doing X you must be doing Y.

9. **Similarity**: An adjective X is similar to an adjective Y.

10. **Attribute**: A noun X is an attribute for which an adjective Y expresses a value.

11. **See also**: This is a relation of relatedness between adjectives.

WordNet version 3.0 used in this thesis. It consists of 147,270 uniques strings of nouns, verbs, adjectives and adverbs. However, many strings are unique within a syntactic category, but are also in more than one syntactic category. Tables 3.1 presents the number of words, synsets and senses for the various parts of speech. Table 3.2 presents the polysemy information while Table 3.3 presents the average polysemy information for the various parts of speech.

Table 3.1 Number of Words, Synsets and Senses in WordNet 3.0

POS	Unique Strings	Synsets	Total Word-Sense Pairs
Noun	117,798	82,115	146,312
Verb	11,529	13,767	25,047
Adjective	21,479	18,156	30,002
Adverb	4,481	3,621	5,580
Total	155,287	117,659	206,941

Table 3.2 Polysemy Information in WordNet 3.0

POS	Monosemous Words and Senses	Polysemous Words	Polysemous Senses
Noun	101,863	15,935	44,449
Verb	6,277	5,252	18,770
Adjective	16,503	4,976	14,399
Adverb	3,748	733	1,832
Total	128,391	26,896	79.450

Table 3.3 Average Polysemy Information in WordNet 3.0

POS	Including Monosemous Words	Excluding Monosemous Words
Noun	1.24	2.79
Verb	2.17	3.57
Adjective	1.40	2.71
Adverb	1.25	2.50

We particularly focus on some of the important relations in WordNet which forms part of implementation of the modified Lesk algorithm used in this work, that had not been previously explored in word sense disambiguation. First, is the derivationally related forms of a word which are generally thought to be highly regular and productive, and the addition of given affixes to their base forms produce new words whose meanings differ from that of the base words in a predictable way. Also used are the morphologically related nouns (pertainyms) of adjectives and adverbs which is applicable to adjectives and adverbs formed from nouns. For instance, the following can be derived from the word *"editorial"*: *"editorialize"*, *"editorialist"*, *"editor"* and the noun form *"voice"* can be derived from the adjective *"vocal"*. The second relation is the antonymy relation which is the opposite of the target word e.g wet – dry. The third is entailment (verbs). According to WordNet glossary, entailment is defined as: "A verb X entails Y if X cannot be done unless Y is, or has been done" e.g divorce entail marry. The last is causes (verbs) and this can be defined as an action or actions (causative) that triggers another action (resultative) e.g. give – have.

3.2.1.3 Semantic Relatedness and Similarity

From the perspective of word sense disambiguation, we distinguish between semantic relatedness and similarity. For a discourse to be coherent, words in the discourse must be related in meaning [42]. This is a natural property of human languages and at the same time one of the most powerful constraints used in automatic word sense disambiguation. Words that share a common context are usually closely related in meaning, and therefore the appropriate senses can be selected by choosing those meanings found within the smallest semantic distance [84]. Semantic similarity metrics use this distance to compute the similarity between two word senses. While this kind of semantic constraint is often able to provide unity to an entire discourse, its scope has been usually limited to a small number of words found in the immediate vicinity of a target word, or to words connected by syntactic dependencies with the target word. These methods target the local context of a given word, and do not take into account additional contextual information found outside a certain window size. Similar to the Lesk algorithm, these similarity methods become extremely computationally intensive when more than two words are involved.

Patwardhan et al [80] distinguish between semantic relatedness and similarity following Budanitsky and Hirst [14]. They described semantic similarity as a kind of relatedness between two words that defines resemblance and semantic relatedness as a broader relationship between concepts that includes similarity as well as other relations such as is-a-kind-of, is-a-part-of etc. as found in WordNet. Word sense similarity measures based on WordNet typically use one form of semantic relation or the other to compute similarity between two word senses using the semantic distance. Semantic similarity measures include but not limited to Wu and Palmer [107], Reisnik [85], Agirre and Rigau [2], Leacock et al. [56], Hirst and St-Onge [43], Lin [61], Mihalcea and Moldovan [69] measures. Next, the Jiang and Conrath measure [46] is discussed, on which part of our hybrid WSD algorithm is based. We have adopted the Jiang and Conrath similarity measure for our final experiment because in the various experiments with each of the semantic similarity measures, we found it to be the best performing similarity measure.

3.2.1.4 The Adapted Lesk Algorithm

The adapted Lesk algorithm [8] takes as input an instance in which a single target word occurs, and will output WordNet sense for that target word based on the information about the target word and a few immediately surrounding words. They defined the context of the target word to be a window of WordNet word tokens n to the left and another n tokens to the right, thereby totalling $2n$ surrounding words. The target word is also included so that for every target word, there is a $2n+1$ word tokens. If the target word is near the beginning or the end of the instance, additional WordNet words is added from the other direction as the case may be.

To disambiguate and return a WordNet sense for a target word, the senses of the target word and that of immediate surrounding ones are considered. Each word W_i has one or more possible senses, each represented by a unique sense tag referring to a particular sense of that word. Using the number of sense tags of the word W_i, denoted as $|W_i|$, an evaluation of each possible combination of sense tag assignments for the context window words is done. Each of such combinations is given by expression $\Pi_{i=1}^{N}|W_i|$, referred to as the candidate combination. A combination score is then computed for each candidate combination. The sense tag that achieves the highest score is chosen as the correct WordNet sense of the target word.

The combination scoring works by comparing glosses between each pair of words in the context window. For instance, if there are N words in the window of context, then there are $N(N-1)/2$ pair of words to be compared. There are a series of relation pairs that identify which synset is to provide the gloss for each word in a pair during comparison. There is a mixture on the choice of which gloss to use in a particular situation. For example, a relation pair might specify that the gloss of a synset of one word be compared with the gloss of the hypernym of the other word. However, the glosses to be compared for each relation pair are those associated with the senses of the candidate combination. The set of relations used for the experiment are hypernyms, hyponyms, troponyms, holonyms, meronyms and attributes of each word in the pair. If the part of speech of a word is known as applicable for single-valued POS words, the synsets and relations are restricted to those associated with that part of speech otherwise, the synsets and relations associated with all the possible part of speech are used.

In comparing the glosses of two context words, an overlap is defined between them to be the highest number of words appearing in both glosses. Each overlap contributes a score equal to the square of the number of words in the overlap. Once all comparions have be done for every pair of words in the window size based on every relation pair, the score of individual scores of the comparisons is added to give the combination score for the particular candidate combination of each sense tags. This is repeatedly done until all candidate combinations have been scored. The candidate combination with the highest score is the winning combination and the target word is assigned to the sense tag given in that combination.

3.2.2 Jiang and Conrath Similarity Measure

Jiang and Conrath similarity [46] is a similarity metric derived from corpus statistics and the WordNet lexical taxonomy. It is a combined model that is derived from the edge-based notion by adding the information content (IC) as a decision factor. The model is based on the lexical taxonomy of the lexicon and statistics in the information content. Jiang and Conrath uses the difference in the information content of the two concepts to indicate their similarity.

For this algorithm, Reisnik's IC measure [85] is augmented with the notion of path length between concepts. This approach includes the information content of the concepts themselves along with the information content of their lowest common subsumer. A lowest common subsumer is a concept in a lexical taxonomy which has the shortest distance from the two concepts compared. They argue that the strength of a child link is proportional to the conditional probability of encountering an instance of the child sense s_i given an instance of its parent sense. The resulting formula can be expressed in Equation 3.1

$$Dist(w_1,w_2) = IC(s_1) + IC(s_2) - 2 \times IC(Lsuper(s_1,s_2)), \tag{3.1}$$

where s_1 and s_2 are the first and second senses respectively and LSuper (lowest common subsumer) is the lowest super-ordinate of s1 and s2. IC is the information content given by equation 3.2.

$$IC(c) = log^{-1}P(s) \tag{3.2}$$

$P(s)$ is the probability of encountering an instance of sense s.

Chapter 4

Topic Modelling and Text Classification

4.1 Topic Modelling

Topic models provide a powerful tool for analyzing large text collections by representing them as a low dimensional set of topics. Topic models [44, 11, 36–38] are based upon the idea that documents are mixtures of topics, where a topic is a probability distribution over words. A topic model is a generative model for documents, that is, it specifies a simple probabilistic procedure by which documents can be derived. Each topic is a multinomial distribution over words and the highest probability words summarize the subjects in the document collection.The major strength of topic models are dimensionality reduction and thematic semantic information extraction. Topic models have been applied in different areas including Information Retrieval, Text Categorization, Sentiment Analysis, Data Mining, Document Summarization etc. The combination of probabilities in a topic gives the topic weight. Representing the documents with probabilistic topics has one distinct advantage over a purely spatial representation, a topic is individually interpretable, providing a probability distribution over words which depicts a cluster of correlated words. In other words, topic models reveal the subject of discussion in documents, for instance in communications like e-mail, tweets etc.

A generative model for documents is based on simple probabilistic sampling rules that describe how words in documents might be generated on the basis of latent variables. When fitting a generative model for a collection, the goal is to find the best set of random variables that can explain the observed words in the documents. The generative process does not make any assumptions about the order of words as they appear in documents. The only information relevant to the model is the number of times words appear in the documents. This is known as the bag-of-words model, and is common in many statistical models of language including LSI and pLSI. However, word order information in some cases might contain important clues to the content of a document. Blei et al. [9] present an extension of the topic model that is sensitive to word order and automatically learns the syntactic as well as semantic factors that guide word choice

Probabilistic models [44, 11, 36–38] use the same idea as generative models – that a document is a mixture of topics, with a slightly different statistical assumptions. We take $P(z)$ for the distribution over topics z in a particular document and $P(w|z)$ for the probability distribution over words w given topics z. Each word w_i in a document, where i refers to the ith word token which is generated by first sampling a topic from the topic distribution, then choosing a word from the topic-word distribution. We denote $P(z_i = j)$ as the probability that the jth topic was sampled for the ith word token and $P(w_i|z_i = j)$ as the probability of word w_i under topic j. The model specifies the following (equation 4.1) distribution over words within a document:

$$P(w_1) = \sum_{j=1}^{T} P(w_i|z_i)P(z_i = j) \tag{4.1}$$

where T is the number of topics, $\phi^{(j)} = P(w|z = j)$ refer to the multinomial distribution over words at index $i(i > 0)$ for topic j and $\theta^{(d)} = P(z)$ is the multinomial distribution over topics for document d. Assuming the document collection consists of D documents and each document d consists of Nd word tokens where N be the total number of word tokens, that is, $N = \sum Nd$. The parameters ϕ and θ respectively, indicate which words are important for which topic and which topics are important for a particular document. Probabilistic Latent Semantic Indexing method (pLSI) [44] marks the introduction of probabilistic topic modelling. However, the pLSI model does not make any assumptions about how the mixture weights θ are generated, making it difficult to test the effectiveness of the model on new documents.

4.1.1 Latent Dirichlet Allocation (LDA) Model

Latent dirichlet allocation (LDA) model [11] extended the pLSI by introducing a dirichlet prior on θ. LDA is both a generative and probabilistic model which models documents in a collection as a finite mixture of latent topics and each topic in turn modelled as a set of topic is characterized by distribution over words. The topics generated by LDA capture correlations among words in which words that have semantic relations belong to the same topic. It basically comprises three-level hierarchical bayesian model. LDA [11] formally defines the following notations and terminologies:

- A word is the basic unit of discrete data, defined to be an item from a vocabulary indexed by $\{1,.....V\}$. Words are represented using unit-basis vectors that have a single component equal to one and all other components equal to zero. Hence using superscripts to denote components, the vth word in the vocabulary is represented by a V-vector w such that $w^v = 1$ and $w^u = 0$ for $u \neq v$.

- A document is a sequence of N words denoted by $w = (w_1, w_2, ...w_N)$, where w_n is the nth word in the sequence.

- A corpus is a collection of M documents denoted by $D = \{w_1, w_2, ..., w_M\}$

According to [11], LDA assumes the following generative process for each document w in a corpus D:

1. Choose $N \sim$ Poisson(ξ).

2. Choose $\theta \sim$ Dirichlet(α).

3. For each of the N words w_n:

   (a) Choose a topic $z_n \sim Multinomial(\theta)$.

   (b) Choose a word w_n from $p(w_n|z_n,\beta)$, a multinomial probability conditioned on the topic z_n.

The basic assumptions of LDA are: the dimensionality k of the Dirichlet distribution and the dimensionality of the topic variable z is assumed known and fixed, the matrix β of size $k \times V$ where $\beta_{ij} = p(w^j = 1|z^i = 1)$ are parameters used for word probabilities and the Poisson assumption is not decisive for anything that follows and more realistic document length distributions can be used as desired. A k-dimensional dirichlet random variable θ can take values in the (k-1)-simplex and has on this simplex, a probability density presented in equation 4.2:

$$p(\theta|\alpha) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1}, \tag{4.2}$$

where the parameter α is a k-vector with components $\alpha_i > 0$, and where $\Gamma(x)$ is the Gamma function. The dirichlet is a convenient distribution on the simplex and has finite dimensional sufficient statistics, and is conjugate to the multinomial distribution. These properties enable inference development and parameter estimation algorithms. Given the parameters α and β, the joint distribution of a topic mixture $theta$, a set of N topics z, and a set of N words w is given by equation 4.3:

$$p(\theta, z, w|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta), \tag{4.3}$$

where $p(z_n|\theta)$ is simply θ_i for the unique i such that $z_n^i = 1$. Integrating over θ and summing over z, the marginal distribution of a document is obtained in equation 4.4:

$$p(w|\alpha, \beta) = \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta) p(w_n|z_n, \beta) \right) d\theta. \tag{4.4}$$

Taking the product of the marginal probabilities of single documents, we obtain the probability of a corpus in equation 4.5:

$$p(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d. \tag{4.5}$$

There are three levels to LDA representation as shown in the LDA probabilistic graphical model presented in Figure 4.1.

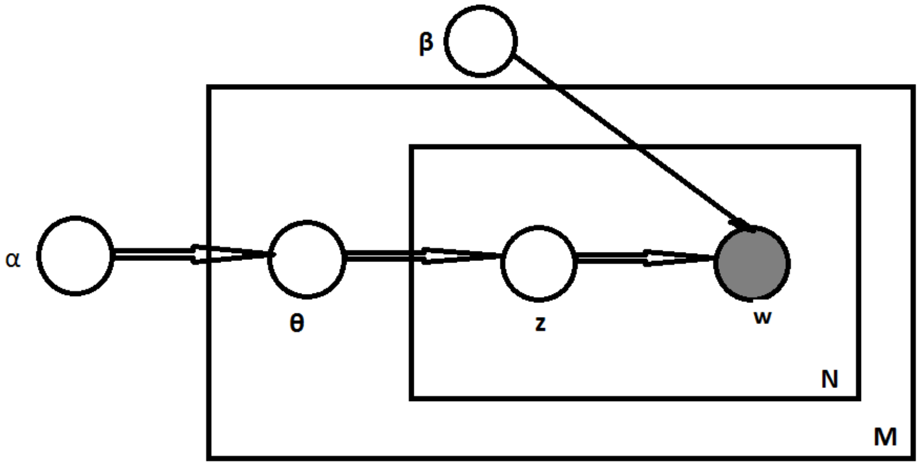

Fig. 4.1 Graphical Representation of LDA Model.

The parameters α and β are collection-level parameters, assumed to be sampled once in the process of generating a collection. The variables θ_d are document-level variables, sampled once per document while the variables z_{dn} and w_{dn} are word-level variables and are sampled once for each word in each document.

In LDA [11], words are assumed generated by topics (fixed conditional distributions) and that those topics are infinitely exchangeable within a document. Using de Finetti's theorem, the probability of a sequence of words and topics must therefore have the form presented in equation 4.6:

$$p(w,z) = \int p(\theta) \left(\prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n) \right) d\theta, \tag{4.6}$$

where θ is the random parameter of a multinomial over topics. LDA distribution is obtained on documents by marginalizing out the topic variables and endowing θ with a dirichlet distribution.

4.1.1.1 LDA as a Mixture of Unigrams

By marginalizing over the hidden topic variable z, LDA can be taken as a two-level model. In particular, let us form the word distribution $p(w|\theta,\beta)$:

$$p(w|\theta,\beta) = \sum_{z} p(w|z,\beta) p(z|\theta). \tag{4.7}$$

$p(w|\theta,\beta)$ is a random quantity since it depends on θ. A new generative process is defined for a document w using the following assumptions:

1. Choose $\theta \sim \text{Dirichlet}(\alpha)$.
2. For each of the N words w_n:
   (a) Choose a word w_n from $p(w_n|\theta,\beta)$.

This process defines the marginal distribution of a document as a continuous mixture distribution of words (a continuous mixture of unigrams) presented in equation 4.8:

$$p(w|\alpha,\beta) = \int p(\theta|\alpha) \left(\prod_{n=1}^{N} p(w_n|\theta,\beta) \right) d\theta, \tag{4.8}$$

where $p(w_n|\theta,\beta)$ are the mixture components and $p(\theta|\alpha)$ are the component weights.

LDA overcomes the problems associated with pLSI by treating the topic mixture weights as a k-parameter hidden *random variable* rather than a large set of individual parameters which are explicitly linked to the training set. LDA is a well-defined generative model and generalizes easily to new documents. Furthermore, LDA model results in $k+kV$ parameters in a k-topic which do not grow with the size of the training corpus; LDA does not suffer from the same overfitting issues as pLSI. In general, LDA posits that each word in both the observed and unseen documents is generated by a randomly chosen topic which is drawn from a distribution with a randomly chosen parameter. This parameter is sampled once per document from a smooth distribution on the topic simplex.

4.1.1.2 Inference in LDA

The key inferential problem that needs to be solved in order to use LDA is that of computing the posterior distribution of the hidden variables, given a document given by equation 4.9:

$$p(\theta,z|w,\alpha,\beta) = \frac{p(\theta,z,w|\alpha,\beta)}{p(w|\alpha,\beta)}. \tag{4.9}$$

This distribution is intractable to compute in general. Although the posterior distribution is intractable for exact inference, a wide variety of approximate inference algorithms can be considered for use with LDA, including Laplace approximation, variational approximation, and Markov chain Monte Carlo. In this thesis, we employ Gibbs sampling, a specific form of Markov chain Monte Carlo to compute the posterior distribution which we discuss in section 4.1.3.

4.1.1.3 Parameter Estimation in LDA

Parameter estimation in the LDA mode is by empirical Bayes method. Giiven a corpus of documents $D = \{w_1, w_2, \ldots, w_M\}$, The goal is to find parameters α and β that maximize the (marginal) log likelihood of the data given in equation 4.10:

$$\ell(\alpha,\beta) = \sum_{d=1}^{M} \log p(w_d|\alpha,\beta). \tag{4.10}$$

As earlier mentioned, the quantity $p(w|\alpha,\beta)$ is not manipulable. However, a variational inference [10] approach provides a tractable lower bound on the log likelihood, a bound which can be maximized with respect to α and β. An approximate empirical Bayes estimates for the LDA model can be found via an alternating variational EM procedure that maximizes a lower bound with respect to the variational parameters γ and ϕ, and then, for fixed values of the variational parameters, maximizes the lower bound with respect to the model parameters α and β.

The variational EM algorithm derivation yields the following iterative algorithm in equation :

1. (E-step) For each document, find the optimizing values of the variational parameters $\{\gamma_d^*, \phi_d^* : d \in D\}$.

2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters α and β. This corresponds to finding maximum likelihood estimates with expected sufficient statistics for each document under the approximate posterior which is computed in the E-step.

These two steps are repeated until the lower bound on the log likelihood converges.

4.1.2 Markov chain Monte Carlo and Gibbs Sampling

Monte Carlo methods are computational techniques that make use of random numbers. Markov chain Monte Carlo (MCMC) refers to a set of approximate iterative techniques designed to sample values from complex (usually high-dimensional) distributions [74, 33, 97]. Markov chain Monte Carlo is a procedure for obtaining samples from complicated probability distributions which allows a Markov chain to converge to a target distribution from which samples are drawn. Gibb sampling is a form of Markov chain Monte Carlo. Gibbs sampling, also known as the heat bath method or 'Glauber dynamics', is a method for sampling from distributions over at least two dimensions.

4.1.3 Gibbs Sampling for Topic Extraction in LDA

We followed the method of Griffiths and Steyverts in extracting topics from documents using Gibbs sampling in LDA [38]. Representing document as a probability distribution over the words. They gave probability of each word in the topic is given by equation 4.11:

$$p(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j), \tag{4.11}$$

where z_i is a latent variable indicating the topic from which the ith word was drawn and $P(w_i|z_i = j)$ is the probability of the word w_i in jth topic. $P(z_i = j)$ is the probability of choosing a word from topics j in the current document, which will vary across different documents. In this case, $T = 1$, therefore $j = 1$. $P(w|z)$ indicates which words are important to the topic. This was the basis for the probabilistic latent semantic indexing (pLSI) and the latent dirichlet allocation model which we discussed in previous sections. Recall that the LDA uses a simple probabilistic procedure by which new documents can be produced given just a set of topics ϕ, allowing ϕ to be estimated without requiring the estimation of θ. In LDA, documents are generated by first picking a distribution over topics θ from a dirichlet distribution, which determines P(z) for words in that document. The words in the document are then generated by picking a topic j from this distribution and then picking a word from that topic according to $P(w|z=j)$, which is determined by a fixed $\phi^{(j)}$ which results in maximizing $P(w|\phi,\alpha)=\int P(w|\theta)P(w|\alpha)d\theta$. This, as earlier mentioned is intractible. The LDA uses a variational inference technique to solve this problem while other techniques such as expectation propagation have been used in other works.

The posterior distribution of words assignment to topics $p(z|w)$ is first done by employing the probability model of LDA with addition of Dirichlet prior on ϕ producing a complete probability model as follows:

$$w_i|z_i \sim Discrete(\phi^{z^i}),$$

$$\phi \sim Dirichlet(\beta),$$

$$z_i|\theta^{d_i} \sim Discrete(\theta^{d_i}),$$

$$\theta \sim Dirichlet(\alpha),$$

where α and β are hyperparameters denoting the nature of the priors on θ and ϕ. The dirichlet priors are assumed to be symmetric, each of them having a single value and are conjugate to the multinomial distributions ϕ and θ. This allows for the computation of joint distribution $p(w,z)$ by integrating out ϕ and θ. Integrating out ϕ gives the term:

$$P(w|z)=\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^{w}}\right)^{T}\prod_{j=1}^{T}\frac{\prod_{w}\Gamma(n_j^{(w)}+\beta)}{\Gamma(n_j^{(.)}+W\beta)}, \tag{4.12}$$

where $n_j^{(w)}$ is the frequency of assignment of word w to topic j in the vector assignments z and $\Gamma(.)$ is the standard gamma function. The second term results from integrating out θ to give equation 4.13:

$$P(z)=\left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^{T}}\right)^{D}\prod_{d=1}^{D}\frac{\prod_{j}\Gamma(n_j^{(d)}+\alpha)}{\Gamma(n.^{(d)}+T\alpha)}, \tag{4.13}$$

where $n_j^{(d)}$ is the frequency of assignment of a word from document d to topic j. The goal is to evaluate the posterior distribution

$$P(z|w)=\frac{P(w,z)}{\sum_{z}p(w,z)}. \tag{4.14}$$

Unfortunately, it is impossible to compute this distribution directly because the sum in the denominator does not factorize and involves T^n terms, where n is the total number of word instances in the corpus. Using the Gibbs sampling where the next state is reached by sequentially sampling all variables from their distribution when conditioned on the current values of other variables. The full conditional distribution $P(z_i|z_i,w)$ is required to use this algorithm which can be obtained by cancellation of terms in expressed in equation 4.15:

$$P(z_i = j|z_{-i},w)\alpha \frac{n_{-i,j}^{(w_i)}+\beta}{n_{-i,j}^{(\cdot)}+W\beta} \frac{n_{-i,j}^{(d_i)}+\alpha}{n_{-i,\cdot}^{(d_i)}+T\alpha}, \tag{4.15}$$

where $n_{-i}^{(\cdot)}$ is a count which does not include the present assignment of z_i. The first ratio expresses the probability of word w_i belonging to topic j and the second ratio expresses the probability of topic j in document d_i. The topic variables z_i are initialized as a sequence on intergers $1,2,...,T$ which determines the initial state of the Markov chain. The Gibbs sampler runs iteratively for fifty iterations using equation 4.15 assigning words to topics until iterations are exhausted and the final topics generated. With samples from the posterior distribution $P(z|w)$ the proportion of individual topics can be computed by integrating the full set of samples and for each single sample, the ϕ and θ can be estimated from the value of z by equations 4.16 and 4.17 respectively:

$$\hat{\phi}_j^{(w)} = \frac{n_j^{(w)}+\beta}{n_j^{(\cdot)}+W\beta}, \tag{4.16}$$

$$\hat{\theta}_j^{(d)} = \frac{n_j^{(d)}+\alpha}{n.^{(d)}+T\alpha}. \tag{4.17}$$

$\phi_j^{(w)}$ and $\theta_j^{(d)}$ corresponds to predictive distribution over new words w and new topics z conditioned on w and z necessary for making inferences on unseen documents.

Chapter 5

Modified Lesk and Hybrid Word Sense Disambiguation Algorithms

5.1 The Modified Lesk Algorithm

Following the adapted Lesk algorithm [8], we introduce a number of modifications to the adapted Lesk algorithm. The modified Lesk algorithm takes as input an instance of the target word and outputs a WordNet sense for the target word based on the information about the target word and all the surrounding words. Experimental evidence shows that by selecting immediate surrounding context words appearing before and after the target words, some important and useful words to sense disambiguation are lost. Therefore, exhaustive consideration of all the surrounding words is necessary for performance and accuracy.

In order to disambiguate and return the correct WordNet sense for a target word, the senses of the target word and that of all the surrounding ones are considered. Each word W_i has one or more possible senses, each represented by a unique sense tag referring to a particular sense of the word. Each of the context words is tagged into part-of-speech using the Stanford tagger [50]. Using the numbered of part-of-speech-based sense tags of the word W_i, denoted as $|W_i|$, an evaluation of each possible combination of sense tag assignments for the context window words is done and each of the combination is given by expression $\Pi_{i=1}^{N}|W_i|$, also called the candidate combination. For the target word, only the sense tags belonging to its part-of-speech are considered. For example, consider the sentence *"Research is key to development"*, which the part-of-speech tagger tags as *('Research', 'NNP'), ('is', 'VBZ'), ('key', 'JJ'), ('to', 'TO'), ('development', 'NN')*. This denotes that the words *research* and *development* are nouns, *is*, a verb and *key*, an adjective. The noun form of the word *research* has two senses, *is* has thirteen senses, the adjective form of *key* has two senses while the development has nine senses. Using the word *research* as the target word and based on the part-of-speech tags of the context words, there are 418 possible combinations on WordNet senses some of which include: research.n.01 $\mapsto be.v.01 \mapsto key.a.01 \mapsto development.n.01, research.n.02 \mapsto be.v.01 \mapsto key.a.01 \mapsto development.n.01, research.n.01 \mapsto be.v.02 \mapsto key.a.01$ $\mapsto development.n.01 etc.$

Prior works in word sense disambiguation using WordNet have not explored the antonymy relation. But according to Ji [45], if two context words are antonymous and belong to the same semantic cluster, they tend to represent alternative attributes for the target word. Furthermore, if two words are antonymous, the gloss and examples of the opposing senses often contain many words that are mutually useful for disambiguating both the original sense and its opposite. Therefore, the glosses of antonyms are used in addition to those of hypernyms, hyponyms, meronyms etc. used by Banerjee and Pedersen [8]. Also, for verbs we include the glosses of entailment and causes relations of each word sense to their vectors. For adjectives and adverbs, we include the morphologically related nouns to the vectors of each word sense in computing the similarity score.

A similarity score is computed for each candidate combination and the sense tag that achieves the highest score is the winning sense for the target word. The similarity score works by comparing glosses between each pair of words in the context window. Each candidate combination consists of combination of glosses of the main senses of the target word together with those of their relations in WordNet, in contrast to the selection of relation pair which supplies the glosses in adapted Lesk method. The glosses of the main senses of target and context words, their examples, glosses of their hypernyms, hyponyms, meronyms, holonyms and antonyms. In the case of verbs, in addition to the aforementioned, also used are the glosses of the entailment and causes. The product is a lemmatized vector of bag of words after the removal of stopwords.

The gloss vector of each sense of the target word, their examples and glosses their relations is compared with the vector of each word sense in each candidate combination. The vectors of individual senses of context words is made up of the glosses of the main sense, their examples and glosses of their relations. A cummulative score is obtained by summing the individual similarity score for each word sense in the candidate combination. The sense of the the target word with highest score is the winning sense for the modified algorithm.

The similarity score (based on maximum overlap) for the modified Lesk algorithm is computed using the cosine similarity, which is the cosine of the angle between the two vectors, which is a measurement of orientation and not magnitude. The magnitude of the score for each word is normalized by the magnitude of the scores for all words within the vector. The resulting normalized scores reflect the degree to which the sense is characterized by each of the component words. Cosine similarity can be computed as the dot product of vectors normalized by their euclidean length given in equation 5.1.

$$\vec{a} = (a_1, a_2, a_3, ...a_n) \quad \text{and} \quad \vec{b} = (b_1, b_2, b_3, ...b_n), \tag{5.1}$$

where a_n and b_n are the components of vectors containing length normalized $tf - idf$ scores for either the words in a context window or the words within the glosses associated with a sense being scored. The dot product is then computed as follows (equation 5.2):

$$\vec{a}.\vec{b} = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + ... + a_n b_n \tag{5.2}$$

The dot product is a simple multiplication of both vectors each having their components summed together. The geometric definition of the dot product given by equation 5.3.

$$\vec{a}.\vec{b} = ||\vec{a}||\ ||\vec{b}||cos\theta \tag{5.3}$$

Using the the cummutative property, we derive equation 5.4.

$$\vec{a}.\vec{b} = ||\vec{b}||\ ||\vec{a}||cos\theta, \tag{5.4}$$

where $||\vec{a}||cos\theta$ is the projection of $\vec{a}$ into $\vec{b}$ in which solving the dot product equation for $cos\theta$ gives the cosine similarity in equation 5.5

$$cos\theta = \frac{\vec{a}.\vec{b}}{||\vec{a}||\ ||\vec{b}||}, \tag{5.5}$$

where $a.b$ is the dot product and $||a||$ and $||b||$ are the vector lengths of a and b, respectively.

5.2 The Hybrid Word Sense Disambiguation Algorithm

A simple illustration of the hybrid algorithm is presented in Figure 5.1.

The text to be disambiguated is split into a set of sentences and the set of sentences split into a set of words called tokens. That is, for a text T, we split T into a set of finite sentences S_i, $S \in T$ and tokenize each sentence S into a set of tokens W_i, $W \in S$. Each token W are then tagged into a part of speech using the stanford part of speech (POS) tagger [50].The single sense of monosemous words are returned as disambiguated based on the part of speech. For polysemous words, the modified Lesk algorithm is applied.

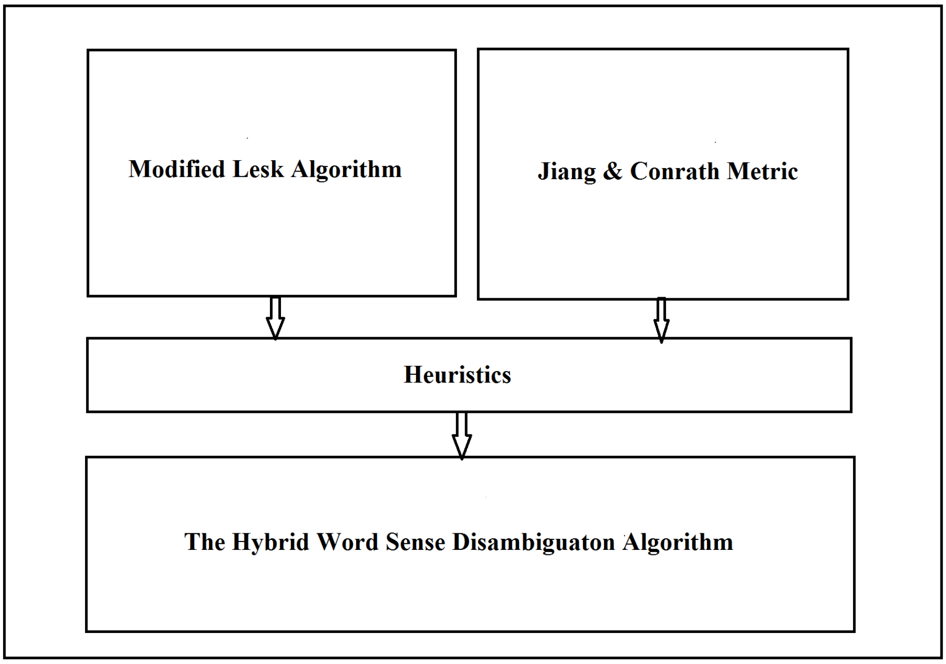

Fig. 5.1 The Hybrid Word Sense Disambiguation Algorithm - A system that combines two distinct disambiguation submodules.

Furthermore, each target word in a sentence is also disambiguated by applying the Jiang and Conrath similarity measure using all the context words as the window size. We did this by computing Jiang and Conrath similarity score for each candidate senses of the target word and select the sense that has the highest cummulative similarity score for all the context words.

For each context word w and candidate word senses c_{eval}, the Jiang and Conrath measure computes individual similarity scores using equation 5.6

$$sim(w, c_{eval}) = \max_{c \in sen(w)}[sim(c, c_{eval})], \tag{5.6}$$

where $sim(w, c_{eval})$ function is the maximum similarity score obtained by computing Jiang & Conrath similarity for all the candidate senses in a context word. The aggregate summation of the individual similarity scores is given in equation 5.7.

$$\text{argmax}_{c_{eval} \in sen(\mathrm{W})} = \sum_{w \in context(\mathrm{W})} \max_{c \in sen(w)}[sim(c, c_{eval})] \tag{5.7}$$

An agreement between the results produced by each of the two algorithms means the target word has been successfully disambiguated and the sense on which they agreed is returned as the correct sense. Whenever one module fails to produce any sense that can be applied to a word but the other succeeds, we just return the sense computed by the successful module. Module failures occur when all of the available senses receive a score of zero according to the module's underlying similarity algorithm (e.g., due to lack of overlapping words for Modified Lesk).

Finally, in a situation where the two modules select different senses, we heuristically resolve the disagreement. Our heuristic first computes the derivationally related forms of all of the words in the context window and adds each of them to the initial vector representation for each of the two competing senses. Then, a similarity score is computed using the cosine similarity between the new vector representations of the two competing senses and vector representation for the context words used initially for all the available word senses. The algorithm returns the sense selected by the module whose winning vector is most similar to the augmented new vectors.

We illustrate the technique discussed above with the first sentence of the Semeval 2007 coarse-grained English all-words data. The target words in this sentence with their parts of speech are: *editorial - noun, ill - adjective, homeless - noun, refer - verb, research - noun, six - adjective, colleague - noun, report - verb, issue - noun*. In this case, we do not have to tokenize and tag into parts of speech since the data is a preprocessed one. The first word in the sentence which is a noun is monosemous because only a single noun sense exists for the word "editorial", therefore the single sense is returned as the right sense. The second word is "ill" and is an adjective and polysemous with adjectival five senses. To disambiguate using the modified Lesk, for each of the senses of *"ill"*, we obtain their glosses and that of their semantically related senses in the WordNet taxonomy. We also obtained the glosses and glosses of semantically related senses of *"editorial", "homeless"," refer", "research", "six", "colleague", "report" and "issue"* and computed the cumulative maximum overlap among them. The sense with the highest cumulative overlap score is returned as the right sense. To disambiguate the other words in the sentence, the same process is applied.

Furthermore, the Jiang and Conrath measure is not applicable to *"ill"* in this case because to compute similarity using Jiang and Conrath measure, the lowest common subsumer(the shortest distance from the two concepts compared) is required. To compute the lowest common subsumer (lcs), the hypernymy relations of the two words being compared is required which adjectives and adverbs do not have. Consequently the modified Lesk algorithm, in this case, is the only producing algorithm for the hybrid algorithm. Using the next word "homeless" to illustrate this, for each of the two noun senses of "homeless", the Jiang and

Conrath similarities for each sense and that of each noun senses of *"editorial", "research", "colleague", and "issue"* are computed. This is because the lowest common subsumer can be computed for senses with the same part of speech. We also compute the cumulative similarity score and returned the sense with the highest score as the right sense of *"homeless"*. Furthermore, the Jiang and Conrath metric described above is applied to the noun words *"editorial", "homeless", "research", "colleague", "issue"* and the verb words *"refer"* and *"report"*.

Finally, the Modified Lesk method returns sense *"homeless%1:14:00::"* as the right sense with a gloss of "someone without a housing" while the Jiang and Conrath method returned *"homeless%1:18:00"* with the gloss *"poor people who unfortunately do not have a home to live in"*. We add the derivationally related forms of each of the words in context of the target word to the original sentence which produces a new vector consisting of the following words: *'editorial', 'editorialize', 'editorialist', 'editor', 'Ill', 'illness', 'Homeless', 'homelessness", 'refer', 'reference', 'referee', 'referral', 'research', 'researcher', 'six', 'colleague', 'collegial', 'reporter', 'report', 'reportage', 'reporting', 'issuer', 'issue', 'issuer', 'issuing'*. In the same manner as being done for the modified Lesk, the glosses each of *"homeless% 1:18:00::"* and *"homeless%1:14:00::"*, that of their hypernyms, hyponyms, meronyms, antonyms etc are computed. The cosine similarity score between their gloss vectors and the new vector is then obtained. The final validation experiment returned *"homeless%1:18:00"* as the right sense. The same technique is applicable to all the target words where the results of modified Lesk and Jiang and Conrath algorithms do not agree.

The intuition behind this notion of validation is that the glosses of a word sense, and that of their semantically related ones in the WordNet lexical taxonomy should share words in common as much as possible with contextual words. Adding the derivationally related forms of the words from the context words increases the chances of overlap when there are mismatches caused by changes in word morphology.

5.2.1 Adaptation of the Hybrid Word Sense Disambiguation Algorithm for Multilingual Application

We adapt the hybrid WSD algorithm to multilingual setting using a multilingual knowledge resource; BabelNet. The adaptation is useful for task such as cross-lingual Information Retrieval and machine translation among others.

5.2.1.1 BabelNet

BabelNet [73] aims at providing an "encyclopedic dictionary" by merging WordNet, described in section 3.2.1.2 and Wikipedia (https://www.wikipedia.org/). Wikipedia is a multilingual web-based encyclopedia. It is a collaborative open source medium edited by volunteers to provide a very large wide-coverage repository of encyclopedic knowledge. Each article in Wikipedia is represented as a page (referred to as Wikipage) and presents information about a specific concept (e.g., Business (Company laws)) or named entity (e.g., Soccer (1991) video game). The title of a Wikipage (e.g., Business (Company laws)) is composed of the lemma of the concept defined (e.g., concern, business, business concern) plus an optional label in parentheses which specifies its meaning if the lemma is ambiguous.

The text in Wikipedia is partially structured. Apart from Wikipages having tables and infoboxes (a special kind of table which summarizes the most important attributes of the entity referred to by a page, such as the history of Soccer (1991) video game), various relations exist between the pages themselves. These include:

- Redirect pages: These pages are used to forward to the Wikipage containing the actual information about a concept of interest. This is used to point alternative expressions for a concept to the same entry, and thus models synonymy.
- Disambiguation pages: These pages collect links for a number of possible concepts through which an arbitrary expression could be referred to.
- Inter-language links: Wikipages also provide links to their counterparts (i.e., corresponding concepts) contained within wikipedia in other languages.
- Categories: Wikipages can be assigned to one or more categories, i.e., special pages used to encode topics.

BabelNet encodes knowledge as a labelled directed graph $G = (V,E)$ where V is the set of nodes – i.e., concepts such as Soccer and named entities such as Soccer (1991) video game and $E \subseteq V \times R \times V$ is the set of edges connecting pairs of concepts. Each edge is labelled with a semantic relation from R, i.e., $\{$is-a, part-of , . . . , $\varepsilon\}, where\ \varepsilon$ denotes an unspecified semantic relation. Each node $v \in V$ contains a set of lexicalizations of the concept for different languages, e.g., $\{$Business$_{EN}$, Importance$_{FR}$, Konzern$_{DE}$, azienda$_{IT}$, asunto$_{ES}\}$. Such multilingually lexicalized concepts are called Babel synsets. Concepts and relations in BabelNet are gathered from the publicly available semantic lexicon of English, WordNet, and a wide-coverage collaboratively-edited encyclopedia, Wikipedia. In order to build the BabelNet graph, the authors collect at different stages:

1. From WordNet, all available word senses (as concepts) and all the lexical and semantic pointers between synsets (as relations);

2. From Wikipedia, all encyclopedic entries (i.e., Wikipages, as concepts) and semantically unspecified relations from hyperlinked text.

To enable multilinguality, BabelNet developers collect the lexical realizations of the available concepts in different languages. Finally, they connect the multilingual Babel synsets by establishing semantic relations between them. Thus, the methodology consists of three main steps:

1. Combination of WordNet and Wikipedia by automatically acquiring a mapping between WordNet senses and Wikipages. This avoids duplicate concepts and allows their inventories of concepts to complement each other.

2. Harvest of multilingual lexicalizations of the available concepts (i.e., Babel synsets) by using (a) the human-generated translations provided by Wikipedia (the so-called interlanguage links), as well as (b) a machine translation system to translate occurrences of the concepts within sense-tagged corpora

3. Establishment of relations between Babel synsets by collecting all relations found in WordNet, as well as all wikipedias in the languages of interest in order to encode the strength of association between synsets, from which their degree of correlation using a measure of relatedness based on the Dice coefficient is computed.

5.2.1.2 The Adapted Hybrid Multinlingual Word Sense Disambiguation Algorithm

The adapted hybrid multilingual is the same as the hybrid algorithm described in section 5.2 except that the wikipedia glosses of relations of the language word being disambiguated and that of the context words are added to the existing vectors. The Jiang and Conrath similarity module remains the same for the monolingual algorithm and it is computed using the information contained in WordNet. This is made possible by the mapping of WordNet to Wikipedia provided in BabelNet.

Chapter 6

The Combined eDiscovery Classification Algorithm

6.1 The Combined Unsupervised Classification Algorithm

In this section, we discuss the novel combination of the word sense disambiguation and topic modelling algorithms making up the combined classification algorithm. An illustration of the combined technique is presented in Figure 5.1 The coloured parts shows a static entity or the end result of a process while the plain parts indicates the processes.

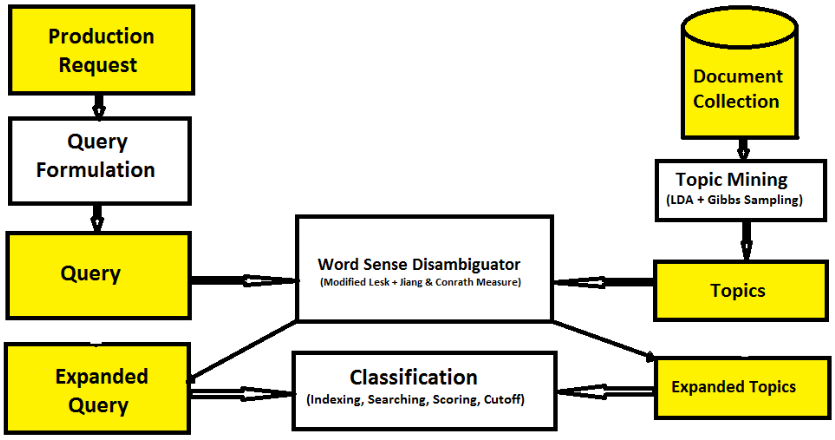

Fig. 6.1 System Diagram of the eDiscovery Classification Techniques.

6.1.1 Production Request

In the search for relevant documents, the first step in the electronic discovery process is the specification and definition of relevance by the parties involved in the litigation or regulatory matter. The production request states all the conditions the relevant documents must meet. We take as an example, the interactive task topic 301 of the TREC 2010 legal track [22]. It states that:
"All documents or communications that describe, discuss, refer to, report on, or relate to onshore or offshore oil and gas drilling or extraction activities, whether past, present or future, actual, anticipated, possible or potential, including, but not limited to, all business and other plans relating thereto, all anticipated revenues therefrom, and all risk calculations or risk management analyses in connection therewith."
According to the relevance guidelines, the target documents must belong to one or more of the following three categories:

1. Any documents related to the business of onshore or offshore oil and gas drilling or extraction activities.
2. Business plans and other documents providing information regarding revenues from oil and gas drilling or extraction activities.
3. Documents reflecting risk calculations or risk management analyses about oil and gas drilling or extraction activities.

It went further to spell out the conditions under which a document can be classified as responsive, divided into four categories; general, category 1, category 2 and category 3. The general category specifies the following:

> "Each document is to be evaluated for responsiveness within the "four corners" of the document. Responsiveness should not be speculative. A document must provide responsive information without requiring the consideration of tangential information. There are no date restrictions that apply to this topic. Each document is to be assessed for responsiveness regardless of its date. Drafts or redlined versions of documents responsive to this topic should also be considered responsive. Legal privileges should not be considered when assessing responsiveness to this topic. All documents responsive to this topic, regardless of which category or categories they fall within, must contain substantive evidence related to the business of or activities surrounding oil and gas drilling or extraction. Documents may relate to the business of oil and gas drilling and extraction anywhere in the world, not just in the United States. Documents responsive to this topic may discuss either or both onshore or offshore oil and gas drilling or

> extraction activities. This would include documents discussing the transportation of oil and/or gas along pipelines. All documents responsive to this topic should relate specifically to activities performed by the company, not just to general industry activity or the business of oil and gas drilling or extraction generally. Documents that discuss oil and gas drilling or extraction activities generally, or as applicable to the industry at large, without specifically referencing activity performed by the company are nonresponsive."

Category 1 specifies the following conditions for responsiveness:

> "Documents that discuss the compan's actual past or present oil and gas drilling or extraction activities are responsive. Documents that discuss the company's planned or anticipated future oil and gas drilling or extraction activities are responsive even if the future activity is not carried out. Documents that discuss possible or potential oil and gas drilling or extraction activities to be undertaken by the company are responsive even if the activity is not carried out."

The conditions under category 2 includes:

> "Documents must discuss revenue derived from oil and gas drilling or extraction activities. Documents may be "high-level" or "company-wide" documents that have, as a component, a discussion of revenue derived from oil and gas drilling or extraction activities. A document may be responsive to this topic even if it covers a much broader report on or discussion about financial information."

Finally for category 3, the condition includes:

> "Documents must involve a risk analysis or risk calculation of the company's oil and gas drilling or extraction activities."

6.1.2 Query Formulation

This is the process of forming a query from a production request. A critical examination of the production requests shows that they cannot be used verbatim. There are always the preamble aspect of the request statement which are useless to the query and needs to be expunged. Therefore, in accordance with TREC legal track tasks which allows for manual query formulation, we reformulate each topic to derive our query. In the topic 301 example, for instance, we removed the general descriptive parts *"All documents or communications that describe, discuss, refer to, report on, or relate to"*, *"whether past, present or future, actual, anticipated, possible or potential, including, but not limited to"* and *"in connection therewith"*. The end result is the required query.

6.1.3 Query

This is the end result of the query formulation. In the example above, the query formulated from the production request is *"onshore or offshore oil and gas drilling or extraction activities, all business and other plans relating thereto, all anticipated revenues therefrom, and all risk calculations or risk management analyses".* The query serves as input to the hybrid disambiguator.

6.1.4 Hybrid Disambiguator

The disambiguator built from the combination of the modified Lesk algorithm and the Jiang & Conrath measure (described in section 5.2). It is used to disambiguate both the query and topics mined from the originalelectronic documents and expands any disambiguated input word in both with their synonyms and keywords in their glosses. The end result of this process is the expanded query and expanded topics which contains the initial terms together with their synonyms and gloss keywords. The expanded forms serve as inputs to the search/classification algorithm.

This expansion is based on the synonyms sets (synsets) of the WordNet lexicon and differs from relevance feedback. Relevance feedback [86] is a local query expansion technique because it uses only a subset of the document set to calculate the set of expansion terms. Thesaurus or lexical expansion is a global query expansion technique because it makes use of the whole document set when calculating the set of expansion terms. A thesaurus or lexicon is a collection of words, where each word has an associated set of words that are related to it.

6.1.5 Document Collection

The collection used for the experiment is the EDRM Enron e-mail dataset version 2 prepared by ZL technologies from the original dataset and has been used for the legal track of TREC since 2009. It is the most popular publicly available dataset for e-mail research derived from the e-mail communications of the defunct Enron corporation. The collection consists of 1.3 million e-mails but there are duplicate messages. For TREC legal track 2010[22] and 2011 [40] tasks on which the evaluation of the combined unsupervised algorithm is carried out, 455, 449 unique messages were identified with 230,143 attachments. Therefore, the full collection used for the tasks and this experiment contains a total of 685,592 documents.

6.1.6 Topic Mining

The latent dirichlet allocation (LDA) [11] model with Gibbs sampling described in sections 4.1.1 and 4.1.3 is used to exract the main topic of discussion from each document. For each document d_i in the collection D, the number of extracted topics, which forms the representative document is exactly one. The Gibbs sampler in Mallet [67] is used to sample each document using fifty (50) iterations. The topics mined from the documents are the inputs to the hybrid disambiguator which generates the expanded topics by adding synonyms and keywords of sense glosses of the words which makes up the initial query.

6.1.7 Topics

Each topic generated from the documents using the topic model (LDA with Gibbs sampling) is a bag of correlated words from the original documents which serves as input to the hybrid disambiguator.

6.1.8 Expanded Query

The query formulated from the production requests and disambiguated with the disambiguator is expanded not only with synonyms of each word sense of the query terms but also keywords in their glosses. Disambiguating the query terms automatically remove all stopwords from the query. It is intuitive to use the keywords in the glosses of chosen senses of the query terms because glosses of disambiguated words contain further hidden collocated and correlated words which enhances terms overlap between queries and documents. For instance, the query derived from topic 301 contains the word *"revenue"*. The hybrid disambiguator determines that the first sense of revenue in WordNet [70, 31] is the correct sense for the word. The synonyms are *"gross"* and *"receipts"* and the gloss is *"the entire of income before deductions are made"*. Tagging the gloss into part-of-speech, we extract the keywords *"entire"*, *"income"*, *"deductions"* and *"made"*. The other words, being stop words are filtered out. The intuition is based on the fact that documents discussing about *"revenue"* apart from the probability of mentioning the word *"revenue"* itself or the synonymous words *"gross"*, or *"receipts"* but there is also, the possibility of the documents containing the related terms such as *"income"* and *"amount"* hidden in the gloss of the term *"revenue"*. This prevents sparsity of matching terms in both queries and documents and also increases distributional semantic similarity for likely relevant documents.

6.1.9 Expanded Topics

The topics mined from the collection and fed into the disambiguator are expanded with synonymous words and sense glosses. These are the final documents derived from the initial documents on which indexing and search/classification are done.

6.1.10 Classification System

The classification system is the final state in the system flow diagram where the documents are classified into two categories; responsive/relevant or nonresponsive/irrelevant. It consists of quite a number of stages. The expanded query derived from the disambiguation of queries is rolled into the indexing part of the system. Indexing involves modelling the documents in an n-dimensional vector space as discussed in chapter 2. There is need for efficient indexing that is fast and effective. Lucene [5], an open source, highly scalable text search engine library available from the Apache Software Foundation is the tool used for this task. When the content of documents are indexed, tokens are first derived from its text value using a process called analysis, and then those tokens are arranged into the index. Indexing involves processing the original data into a highly efficient cross-reference lookup in order to facilitate rapid searching [68]. Analysis involves much more than tokenizing text before being rolled into the index. There are quite other tasks such as lowercasing, obtaining the root form of tokens (normalization e.g lemmatization, stemming) among others. Stopwords have been automatically removed by disambiguating the terms in both query and documents while the Porter stemming algorithm [83] was used for normalization.

To search the index for matching documents, the query terms were also normalized using the Porter stemming algorithm [83]. One of the most important stage of the classification is the scoring. Each documents are scored based on the $tf-idf$ scoring technique discussed in chapter 3. Lucene [5] employs a slightly modified method based on the $tf-idf$ technique. Lucene combines both the boolean and vector space models of information retrieval. The boolean model is the first pass of the scoring process. In this case, the additive operator "OR" is used to join all the expanded query terms. This implies that at first pass search, the system retrieves all documents matching any of the expanded query terms. It thereafter computes terms-documents similarity using the vector space model.

Recall the cosine similarity score:

$$score(q,d) = \frac{\vec{V}(q).\vec{V}(d)}{|\vec{V}(q)|\,|\vec{V}(d)|}$$

V(q) · V(d) being the dot product of the weighted vectors of the query and documents, and $|V(q)|$ and $|V(d)|$ are their normalized euclidean length. For search quality and usability, lucene has refined the vector space model scoring using the following simplifying assumptions of a single field index:

- Normalizing V(d) to the unit vector is known to be problematic in that it removes all document length information. For some documents, removal this information is good, e.g. a document made by duplicating a certain paragraph several times, especially if that paragraph is made of distinct terms. But for documents which contain no duplicated paragraphs, this might be wrong. To avoid this problem, a different document length normalization factor is used, which normalizes to a vector equal to or larger than the unit vector; doc-len-norm(d).

- At indexing, users can specify that certain documents are more important than others, by assigning a document boost. For this, the score of each document is also multiplied by its boost value, doc-boost(d).

- Lucene is field based, hence each query term applies to a single field, document length normalization is by the length of the particular field, and in addition to document boost there are also document fields boosts.

- The same field can be added to a document several times during indexing, and so the boost of that field is the multiplication of the boosts of the separate additions (or parts) of that field within the document.

- At search time, users can specify boosts to each query, sub-query, and each query term, hence the contribution of a query term to the score of a document is multiplied by the boost of that query term; query-boost(q).

- A document may match a multi-term query without containing all the terms of that query (this is correct for some of the queries), and users can further reward documents matching more query terms through a coordination factor, which is usually larger when more terms are matched; coord-factor(q,d).

From above assumptions, the conceptual scoring formula can be derived as presented in equation 6.1:

$$\text{score(q,d)} = \text{coord-factor(q,d)} \cdot \text{query-boost(q)} \cdot \frac{V(q) \cdot V(d)}{|V(q)|} \cdot \text{doc-len-norm(d)} \cdot \text{doc-boost(d)} \tag{6.1}$$

From this conceptual formula, lucene derives the practical scoring formula with the following assumptions that some parameters are computed in advance:

- Query-boost for the query (actually for each query term) is known when search starts.
- Query Euclidean norm, |V(q)| can be computed when search starts, as it is independent of the document being scored. From search optimization perspective, it is a valid question why bother to normalize the query at all, because all scored documents will be multiplied by the same |V(q)|, and hence documents ranks (their order by score) will not be affected by this normalization. There are two good reasons to keep this normalization: (a) Cosine Similarity can be used to find how similar two documents are. One can use lucene for clustering, and use a document as a query to compute its similarity to other documents. In other words, scores of a document for two distinct queries should be comparable. Other applications may also require this and this is what normalizing the query vector, V(q) provides. (b) Applying query normalization on the scores helps to keep the scores around the unit vector, hence preventing loss of score data because of floating point precision limitations.
- Document length norm doc-len-norm(d) and document boost doc-boost(d) are known at indexing time. They are computed in advance and their multiplication is saved as a single value in the index; norm(d).

The practical scoring formula is given by equation 6.2:

$$\text{score(q,d)} = \text{coord(q,d)} \cdot \text{queryNorm(q)} \cdot \sum_{\text{t in q}} (\text{tf(t in d)} \cdot idf(t)^2 \ \cdot \ \text{tgetBoost()} \cdot \text{norm(t, d)} \tag{6.2}$$

where

- tf(t in d) is the term's frequency, defined as the number of times term t appears in the currently scored document d. Documents that have more occurrences of a given term receive a higher score. Note that tf(t in q) is assumed to be 1 and therefore it does not appear in the equation. The tf(t in d) value is given by equation 6.3:

$$\text{tf(t in d)} = frequency^{1/2} \tag{6.3}$$

- idf(t), the inverse of the number of documents in which the term t appears, is presented in equation 6.4:

$$idf(t) = 1 + log(\frac{\text{number of documents}}{\text{documents frequency} + 1}) \tag{6.4}$$

- coord(q,d) is a score factor based on how many of the query terms are found in the specified document. Typically, a document that contains more of the query's terms will receive a higher score than another document with fewer query terms.

- queryNorm(q) is a normalizing factor used to make scores between queries comparable. This factor does not affect document ranking since all ranked documents are multiplied by the same factor, but rather just attempts to make scores from different queries comparable. It is presented in equation 6.5:

$$\text{queryNorm(q)} = \frac{1}{\text{sum of squared weights}^{1/2}} \tag{6.5}$$

  sum of squared weight is calculated using equation 6.6:

$$\text{sum of squared weights} = \text{q} \cdot \text{getBoost()}^2 \sum_{\text{t in q}} (idf(t) \;\cdot \text{t} \cdot \text{getBoost()}^2 \tag{6.6}$$

- $\text{t} \cdot$ getBoost() is a search time boost of term *t* in the query *q* as specified in the query text.

- norm(t,d) encapsulates a few (indexing time) boost and length factors. The lengthNorm is computed when the document is added to the index in according to the number of tokens of this field in the document, so that shorter fields contribute more to the score.LengthNorm is computed by the similarity class in effect at indexing. norm(t,d) is calculated using equation 6.7:

$$\text{norm(t,d)} = \text{lengthNorm} \;\cdot \Pi_{\text{field f in d named as t}}\text{f} \cdot \text{boost()} \tag{6.7}$$

For this experiment, no boosting whatsoever, either on query or during indexing is used. Documents are classified by determining an optimal cut-off point using the computed scores.

6.2 Experimental Setting and Observations

For the actual experiment with the combined unsupervised algorithm, disambiguation of both queries and documents are carried out by considering all the possible parts-of-speech for each word because production requests have reformulated as queries in which the syntactic structure in the original production requests have been removed. Extracted topics from documents also do not have syntactic structure, a necessary condition for accurate part-of-speech tagging. Topic mining from original documents using the LDA model with

Gibbs sampling takes about 48 hours when the dataset is split into three and allowed to run concurrently using the same program module. Each extracted topic consists of words ranging between 0 and 15 depending on the length of the original document. Word sense disambiguation on the representative corpus consistng of all the representative documents takes about 5 weeks when the corpus is split into 10 and allowed to run concurrently using the hybrid disambiguator. It is estimated that to run the disambiguation algorithm on all the documents in the original collection will take about 60 weeks using the same machine and program. The hardware environment under which the experiment was carried out is as follow:

- Processor: Intel(R) Core(TM) i5-4200(U) CPU 1.60GHz 2.30GHz
- System Type: 64-bit Operating System, x64-based processor
- Random Access Memory (RAM): 12GB
- Disk Space Capacity: 1 terabyte

6.2.1 Experimental Output

The uncombined unsupervised algorithm searches the collection and classifies documents into either relevant or not relevant using a cutoff point. The documents are ranked in order of similarity score computed using the vector space model. For the learning task requires the probability estimate of responsiveness between 0 and 1 in the ranked output. The probability estimate is computed using equation 6.8:

$$P(R) = 1 - \frac{r}{L} \tag{6.8}$$

where P(R) is the probability of responsiveness, r is the rank of the document and L is the total number of retrieved documents.

For the interactive task, the ouput is simply a classification of relevant documents at a specified cutoff point determined using the similarity score computed by the vector space model.

Chapter 7

Empirical Evaluation and Results

7.1 Empirical Evaluation of the Hybrid Word Sense Disambiguation Algorithm

The evaluation setting is the SemEval 2007 coarse-grained English all-words task [72]. The task required participating systems to annotate open-class words (i.e. nouns, verbs, adjectives, and adverbs) in a test corpus with the most appropriate sense from a coarse-grained version of the WordNet sense inventory. The test dataset consists of 5,377 text taken from five articles covering the fields of journalism, book review, travel, computer science and biography out of which 2,269 words were annotated for disambiguation. The inter-annotator agreement stands at 93.80% derived from the pairwise agreement of a two-stage annotation of independent annotators. A breakdown of the test dataset consists of 1,108 nouns, 591 verbs, 362 adjectives and 208 adverbs.

We present the results, each for the component algorithms and the hybrid WSD algorithm in Table 7.1:

Note that Jaing & Conrath similarity measure does not have results for adjectives and adverbs because the lowest common subsumer(the shortest distance from the two concepts compared) cannot be computed in WordNet. To compute the lowest common subsumer (lcs), the hypernymy relations of the two words being compared is required which adjectives and adverbs do not have. Consequently, calculating the Jiang & Conrath similarity score depends on the lowest common subsumer.

Table 7.2 presents a juxtaposition of the overall performance of the hybrid algorithm with state-of-the-art WSD algorithms. In evaluating a word sense disambiguation algorithm, coverage is usually an important factor because it affects recall and consequently the F1 metrics. For systems that do not have a full coverage, back-off strategy is often used to supplement the results. It is a strategy where the most frequent sense (MFS) of the target words are supplied as the right sense wherever an algorithm could not produce result. However, backoff strategy has a drawback in that its usage is unpredictable on the performance of the system depending on the dataset. In other words, systems which have full coverage and are independent of back-off are better off. In real world applications, dependence of word sense disambiguation systems on this strategy may enhance or degrade performance depending on the peculiarity of the dataset.

TKB-UO [4] was the best performing unsupervised system at the SemEval 2007 in the coarse-grained English all-words task. Treematch [18], ExtLesk [81] and Degree [81] emerged after the workshop. Degree incorporates weak supervision in its implementation. Hybrid is our hybrid knowledge-based method. ExtLesk without backoff for nouns has precision of 82.7, recall of 69.2 and F1 of 75.4 while Degree without backoff for nouns has a precision of 87.3, recall of 72.7 and F1 of 79.4. The overall results of both ExtLesk and Degree without backoff are not made available.

7.2 Empirical Evaluation of the Combined Unsupervised Classification Algorithm

The evaluation setting for retrieval of electronic documents search/classification for eDiscovery is the legal track of the Text Retrieval Conference (TREC). TREC is an annual evaluation conference which focuses on different information retrieval research areas including web, cross language, enterprise, legal, question answering etc. We focus on the last two most recent of the legal track, which deals with discovery and retrieval of electronically stored information for litigations and regulatory matters. The test collection is the Enron e-mail dataset described in chapter 5. The evaluation is based on the interactive task of 2010 legal track and the learning tasks for both 2010 and 2011 exercises [22, 40], the two being the most recent of the legal track.

7.2.1 TREC Legal Track Learning Task

The learning task of the TREC legal track [22, 40] represents a scenario in which a preliminary search has been carried out on the collection through ad hoc means and a set of documents have been labelled as relevent or not relevant, called the seed set. This set is then used as input to a search process involving humans and/or technology to estimate the probability of relevance of each remaining documents in the collection. The Learning task models the use of automated or semi-automated methods to guide review strategy for a multi-stage document review effort, organized as follows:

1. **Preliminary search and assessment**. The responding party to a request analyzes the production request. Using ad hoc methods, the team identifies a seed set of potentially responsive and nonresponsive documents.

2. **Learning by example**. A learning method is used to rank the documents in the collection from most to least likely to be responsive to the production request, and to estimate the likelihood of responsiveness for each document. The input to the learning method consists of the seed set, the assessments for the seed set, and the unranked collection; the output is a ranked list consisting of the document identifier, the rank of each document and a probability of responsiveness for each document in the collection.

   The two learning objectives; ranking and estimating the likelihood of responsiveness – may be accomplished by the same method or by different methods. Either may be automated or manual. For example, ranking may be done using an information retrieval method or by human review. Estimation may be done in the course of ranking or, for example, by sampling and reviewing documents at representative ranks.

3. **Review Process**. A review process may be conducted, with strategy guided by the ranked list. One possible strategy is to review documents in order, from the most likely to the least likely to be responsive, thereby discovering as many responsive documents as possible for a given amount of effort. Another possible strategy is triage: to review only mid-ranked documents, deeming, without further review, the top-ranked ones to be responsive, and the bottom-ranked ones to be non-responsive.

   The main questions to be answered for the review strategy are: (1) Of a set of documents, how many are responsive and how many are not? (2) What is the probability of each document in the set being responsive? An answer to the first questions gives answer to the second, since the number of responsive documents is determined by the summation of probabilities of individual documents.

For the 2011 learning exercise, assessment for relevance was carried out by four review companies which volunteer the services of their reviewers while that of 2010 was carried by three independent volunteer reviewers. In both, the opinions, agreements and adjudged disagreements form the basis of the gold standard of relevance.

The systems for the tasks learn based on the annotated examples provided by the seed set to rank documents in the collection, at specific cutoffs using an estimate of the relevance likelihood of each document, from the most relevant to the least relevant. These probability estimates and ranks are then used to estimate the precision, recall and F1. Specifically, the probabilities of the ranked documents at a particular cutoff c, are summed to yield an estimate of the number of relevant documents, denoted Rel_c. In addition, the probabilities of all documents in the collection are summed to yield an estimate of the number of responsive documents in the collection, denoted Rel. From these estimates, the recall is computed using equation 7.1:

$$Recall = \frac{\mathrm{Rel}_c}{\mathrm{Rel}} \tag{7.1}$$

Precision is calculated using equation 7.2:

$$Precision = \frac{\mathrm{Rel}_c}{c} \tag{7.2}$$

And finally, F1 computed using equation 7.3:

$$F1 = \frac{2}{\frac{\mathrm{Rel}}{\mathrm{Rel}_c} + \frac{c}{rel\,c}} \tag{7.3}$$

The participants were therefore, required to produce a ranked list of documents at particular cutoffs c, with the probability estimates of each document so ranked. For the 2011 exercise, the various representative cutoffs for the task are 2,000, 5,000, 20,000, 50,000, 100,000 and 200,000. Furthermore, as part of the submission phases, participating teams were required to submit an initial set of probability estimates prior to requesting any responsiveness determinations from the Topic Authority. Following the initial submission, teams were entitled to receive up to 100 responsiveness determinations before being required to submit an interim set of probability estimates. After submitting the first interim results, teams were entitled to receive up to 200 further responsiveness determinations before submitting a second interim set of results. Thereafter, teams were entitled to receive up to 700 additional responsiveness determinations. In total, each team was allowed to request at most 1,000 responsiveness determinations per topic, subject to submitting the required initial and interim results. Each team was required to submit a final set of probability estimates once it had received all the responsiveness determinations requested by the team. In a final mopup phase, all responsiveness determinations requested by all teams were distributed to all teams, who had the opportunity to submit a mopup set of probability estimates. Thus, the final submission used only relevance determinations for documents specified by the submitting team, while the mopup submission used relevance determinations for documents specified by all teams. The 2010 exercise, on the other hand, uses cutoffs at 20,000, 50,000, 100,000 and 200,000.

Participants were asked to declare each run to be automatic or technology-assisted. Automatic runs were allowed to use manual query formulation, but human review of the document collection (other than that provided by TREC via responsiveness determinations) was not permitted. Technology-assisted runs were allowed to avail themselves of any amount of human review. Participants were asked to state the number of hours spent configuring the system, searching the dataset, reviewing documents, and analyzing the results.

The goal of the learning task is to produce all and only documents responsive to a production request in accordance with the requirement in civil litigations. This task is not a real model of search in electronic discovery scenario. It evaluates based on ranking and probability estimation, though systems do use machine learning techniques usually peculiar to project-based approaches. The issue of ranking and estimation do not arise as a document is either relevant or not relevant; a purely classification task. However, it may be useful for determining optimal cutoff point.

7.2.1.1 Topics for the 2011 Learning Task

The 2011 learning task consists of three topics (topics 401, 402 and 403), each topic having designated topic authorities with whom participants can liase with regarding responsiveness determinations.

Topic 401 states the description of what qualifies a document as relevant:

"All documents or communications that describe, discuss, refer to, report on, or relate to the design, development, operation, or marketing of enrononline, or any other online service offered, provided, or used by the Company (or any of its subsidiaries, predecessors, or successors-in-interest), for the purchase, sale, trading, or exchange of financial or other instruments or products, including but not limited to, derivative instruments, commodities, futures, and swaps".

Topic 402 stipulates the relevance description as follows:

"All documents or communications that describe, discuss, refer to, report on, or relate to whether the purchase, sale, trading, or exchange of over-the-counter derivatives, or any other actual or contemplated financial instruments or products, is, was, would be, or will be legal or illegal, or permitted or prohibited, under any existing or proposed rule(s), regulation(s), law(s), standard(s), or other proscription(s), whether domestic or foreign".

While topic 403 states that:

"All documents or communications that describe, discuss, refer to, report on, or relate to the environmental impact of any activity or activities undertaken by the Company, including but not limited to, any measures taken to conform to, comply with, avoid, circumvent, or influence any existing or proposed rule(s), regulation(s), law(s), standard(s), or other proscription(s), such as those governing environmental emissions, spills, pollution, noise, and/or animal habitats".

The seed set for each of the three topics for the 2011 learning task is presented in Table 7.3. The estimated number of responsive documents for each of the topics is presented in Table 7.4:

7.2.1.2 Topics for the 2010 Learning Task

The 2010 learning exercise consists of eight topics out of which we experimented with seven. The topics ranges from 200 to 207. Topic 200 does not have background complaint for production request like others, though it had seven sentences of guidelines on what is relevant and what is not, given to the participants [98]. The following are the relevance descriptions of topics 201 to 207:

- 201: "All documents or communications that describe, discuss, refer to, report on, or relate to the Company's engagement in structured commodity transactions known as "prepay transactions"."

- 202: "All documents or communications that describe, discuss, refer to, report on, or relate to the Company's engagement in transactions that the Company characterized as compliant with FAS 140 (or its predecessor FAS 125)."

- 203: "All documents or communications that describe, discuss, refer to, report on, or relate to whether the Company had met, or could, would, or might meet its financial forecasts, models, projections, or plans at any time after January 1, 1999."

- 204: "All documents or communications that describe, discuss, refer to, report on, or relate to any intentions, plans, efforts, or activities involving the alteration, destruction, retention, lack of retention, deletion, or shredding of documents or other evidence, whether in hard-copy or electronic form."

- 205: "All documents or communications that describe, discuss, refer to, report on, or relate to energy schedules and bids, including but not limited to, estimates, forecasts, descriptions, characterizations, analyses, evaluations, projections, plans, and reports on the volume(s) or geographic location(s) of energy loads."

Table 7.1 Analysis of WSD Results for Component and Hybrid Algorithms by Part of Speech

	Precision				Recall				F1			
Algorithm	Noun	Verb	Adjective	Adverb	Noun	Verb	Adjective	Adverb	Noun	Verb	Adjective	Adverb
Modified Lesk	77.51	66.27	77.90	68.75	76.53	65.82	77.90	68.75	77.04	66.04	77.90	68.75
Jiang & Conrath	76.75	66.26	-	-	76.26	64.81	-	-	76.50	65.53	-	-
Hybrid	82.58	66.26	77.90	68.75	82.58	66.26	77.90	68.75	82.58	66.26	77.90	68.75

Table 7.2 Overall Performance Results of the Hybrid Algorithm in Comparison with State-of-the-art Algorithms

Algorithm	**Precision**	**Recall**	**F1**
TKB-UO	70.207	70.207	70.207
Treematch	73.650	73.650	73.650
Hybrid	76.509	76.509	76.509
ExtLesk(with backoff)	79.1	79.1	79.1
Degree(with backoff)	81.7	81.7	81.7

Table 7.3 Seed Set for 2011 Learning Task

Topic	**Responsive**	**Nonresponsive**	**Total**
401	1,040	1,460	2,500
402	238	1,864	2,102
403	245	1,954	2,199

Table 7.4 Estimated Number of Responsive Documents for each Topic in the 2011 Learning Task

Topic	**Number of Responsive Documents**	**95% Confidence Interval**
401	20,017	(14,595-25,439)
402	3,012	(1,436-4,588)
403	1,239	(166-2,312)

- 206: "All documents or communications that describe, discuss, refer to, report on, or relate to any discussion(s), communication(s), or contact(s) with financial analyst(s), or with the firm(s) that employ them, regarding (i) the Company's financial condition, (ii) analysts' coverage of the Company and/or its financial condition, (iii) analysts' rating of the Company's stock, or (iv) the impact of an analyst's coverage of the Company on the business relationship between the Company and the firm that employs the analyst."
- 207: "All documents or communications that describe, discuss, refer to, report on, or relate to fantasy football, gambling on football, and related activities, including but not limited to, football teams, football players, football games, football statistics, and football performance."

The statistics of the seed set for each of the seven topics for the 2010 learning task is presented in Table 7.5. The estimated number of responsive documents for each of the topics is presented in Table 7.6.

7.2.1.3 Results of the Combined Unsupervised Algorithm for the Learning Tasks

Table 7.7 presents the result combined unsupervised approach on the 2011 Learning Task. Table 7.8 and 7.9 presents the results of stem-based search and topic-based approaches respectively.

In Table 7.10, we present a juxtaposition of the results of the combined unsupervised algorithm with other task participated systems for topic 401. There are more systems not included which participated only in the mop up task. We have included only systems which participated in the final task only. CUTACe (Combined Unsupervised Technique for Automatic Classification in eDiscovery) is the system produced from the combined unsupervised technique. STEMIR and TOPICIR are stem-based and topic-based retrieval components of the combined algorithm. Table 7.11 presents a comparison between participated runs for the final task and the combined unsupervised technique, stem-based search and topic-based retrieval for topic 402. Topic 403 results comparison with participated runs for the final task is presented in Table 7.12 .

Next, table 7.13 presents the performance results for the 2010 learning task. Table 7.14 and 7.15 presents the results of stem-based search and topic-based retrieval respectively.

Table 7.16 through to 7.19 present performance comparisons of recall metric among the participated supervised systems and the combined unsupervised technique at various cutoffs.

Table 7.5 Statistics of Seed Set used for 2010 Learning Task

Topic	Responsive	Nonresponsive	Total
201	168	523	691
202	1,006	403	1,409
203	67	892	959
204	59	1,132	1,191
205	333	1,506	1,839
206	19	336	355
207	80	511	591

Table 7.6 Estimated Number of Responsive Documents for each Topic in the 2010 Learning Task

Topic	Number of Responsive Documents	95% Confidence Interval
201	1,886	(1,181 - 2,591)
202	6,312	(3,793 - 8,832)
203	3,125	(2,069 - 4,180)
204	6,362	(2786 - 9937)
205	67,938	(53563 - 82313)
206	866	(439 - 1293)
207	20,929	(16256 - 25603)

Table 7.7 Results of the combined unsupervised algorithm on 2011 learning task

	2,000			5,000			20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
401	5	3	4	54	19	28	74	5	10	-	-	-	-	-	-	-	-	-
402	63	2	3	75	1	2	91	0	1	-	-	-	-	-	-	-	-	-
403	4	1	1	99	7	14	100	2	3	-	-	-	-	-	-	-	-	-

Table 7.8 Results of the stem-based search on 2011 learning task

	2,000			5,000			20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
401	6	72	11	76	27	40	77	6	11	-	-	-	-	-	-	-	-	-
402	75	2	4	95	1	2	-	-	-	-	-	-	-	-	-	-	-	-
403	99	13	22	100	10	18	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.9 Results of topic-based retrieval on 2011 learning task

	2,000			5,000			20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
401	80	98	88	62	15	24	63	3	6	-	-	-	-	-	-	-	-	-
402	87	2	4	96	1	2	-	-	-	-	-	-	-	-	-	-	-	-
403	4	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.10 2011 Learning Task Result Comparison Table for Topic 401 Showing Recall, Precision and F1 at Representative Cutoffs

	2,000			5,000			20,000			50,000			100,000			200,000		
Runs	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
ISILRFTF	4	35	7	15	61	24	17	17	17	16	6	9	18	4	6	33	3	6
ISIROTTF	10	98	18	10	38	15	12	12	12	11	5	7	12	2	4	27	3	5
ISITRFTF	7	68	12	15	65	25	42	43	42	56	22	32	69	14	23	71	7	13
mlbclsAF	2	17	3	6	23	9	18	18	18	29	12	17	45	9	15	53	5	10
mlblrnTF	3	33	6	9	35	14	14	14	14	14	6	8	13	3	4	18	2	3
rec03TF	7	68	12	18	75	29	50	51	50	66	26	37	82	16	27	89	9	17
tcdAF	9	99	17	6	23	9	11	11	11	29	12	17	42	9	14	77	8	14
USFDSETF	3	27	5	3	13	5	49	48	48	54	21	30	58	12	19	65	7	12
USFEOLTF	9	86	16	17	62	27	30	29	30	36	14	21	44	9	15	51	5	9
USFMOPTF	6	61	11	9	37	15	43	40	41	57	22	32	60	12	20	61	6	11
UWABASAF	9	97	17	9	37	15	24	24	24	46	18	26	54	11	18	58	6	11
UWALINAF	1	12	2	5	20	8	10	10	10	21	9	12	42	8	14	67	7	12
UWASNAAF	6	64	12	8	31	12	22	21	22	38	15	22	38	8	13	38	4	7
CUTACe	5	3	4	54	19	28	74	5	10	-	-	-	-	-	-	-	-	-
STEMIR	6	72	11	76	27	40	77	6	11	-	-	-	-	-	-	-	-	-
TOPICIR	80	98	88	62	15	24	63	3	6	-	-	-	-	-	-	-	-	-

Table 7.11 2011 Learning Task Result Comparison Table for Topic 402 Showing Recall, Precision and F1 at Representative Cutoffs

	2,000			5,000			20,000			50,000			100,000			200,000		
Runs	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
mlbclsAF	3	4	4	4	2	3	13	2	3	46	3	5	74	2	4	88	1	3
rec03TF	51	77	61	57	35	44	75	12	21	88	6	10	88	3	5	88	1	3
tcdAF	2	3	2	7	4	5	32	5	8	62	4	7	76	2	5	100	1	3
UWABASAF	11	16	13	14	8	10	33	5	9	37	2	4	37	1	2	37	1	1
UWALINAF	4	6	5	6	4	5	24	4	6	30	2	3	46	1	3	85	1	3
UWASNAAF	8	12	10	11	6	8	29	4	8	27	2	3	31	1	2	63	1	2
CUTACe	63	2	3	75	1	2	91	0	1	-	-	-	-	-	-	-	-	-
STEMIR	75	2	4	95	1	2	-	-	-	-	-	-	-	-	-	-	-	-
TOPICIR	87	2	4	96	1	2	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.12 2011 Learning Task Result Comparison Table for Topic 403 Showing Recall, Precision and F1 at Representative Cutoffs

	2,000			5,000			20,000			50,000			100,000			200,000		
Runs	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
ISILrFTF	5	3	4	6	2	2	7	0	1	9	0	0	23	0	1	27	0	0
ISIRoTTF	6	4	5	6	2	3	8	0	1	9	0	0	23	0	1	27	0	0
ISITrFTF	8	5	6	12	3	5	47	3	5	49	1	2	60	1	1	63	0	1
mlbclsAF	2	1	1	2	0	1	3	0	0	4	0	0	4	0	0	4	0	0
rec03TF	31	19	23	95	26	41	97	7	12	99	3	5	99	1	2	100	1	1
tcdAF	13	8	10	22	5	9	31	2	4	37	1	2	70	1	2	99	1	1
UWABASAF	17	11	13	22	6	9	59	4	7	62	2	3	61	1	1	62	0	1
UWALINAF	1	1	1	2	0	1	5	0	1	38	1	2	46	1	1	55	0	1
UWASNAAF	14	9	11	46	11	18	53	3	6	53	1	3	59	1	1	64	0	1
CUTACe	4	1	1	99	7	14	100	2	3	-	-	-	-	-	-	-	-	-
STEMIR	99	13	22	100	10	18	-	-	-	-	-	-	-	-	-	-	-	-
TOPICIR	4	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.13 Results of the combined unsupervised algorithm on 2010 learning task

	20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
201	97.6	0.6	1.2	-	-	-	-	-	-	-	-	-
202	100.0	0.3	0.6	-	-	-	-	-	-	-	-	-
203	99.4	1.8	3.5	-	-	-	-	-	-	-	-	-
204	98.0	1.5	3.0	-	-	-	-	-	-	-	-	-
205	97.8	18.4	31.0	-	-	-	-	-	-	-	-	-
206	88.9	0.0	0.0	-	-	-	-	-	-	-	-	-
207	71.1	2.7	5.2	-	-	-	-	-	-	-	-	-

Table 7.14 Results of stem-based retrieval on 2010 learning task

	20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
201	-	-	-	-	-	-	-	-	-	-	-	-
202	-	-	-	-	-	-	-	-	-	-	-	-
203	99.7	2.0	3.9	-	-	-	-	-	-	-	-	-
204	-	-	-	-	-	-	-	-	-	-	-	-
205	-	-	-	-	-	-	-	-	-	-	-	-
206	-	-	-	-	-	-	-	-	-	-	-	-
207	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.15 Results of topic-based retrieval on 2010 learning task

	20,000			50,000			100,000			200,000		
Topics	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
201	-	-	-	-	-	-	-	-	-	-	-	-
202	-	-	-	-	-	-	-	-	-	-	-	-
203	-	-	-	-	-	-	-	-	-	-	-	-
204	-	-	-	-	-	-	-	-	-	-	-	-
205	-	-	-	-	-	-	-	-	-	-	-	-
206	-	-	-	-	-	-	-	-	-	-	-	-
207	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.16 2010 Learning Task Recall Comparison at 20,000 Cutoff

	Topics						
Runs	201	202	203	204	205	206	207
xreLogA	47.2	82.7	60.5	29.8	27.4	49.4	87.5
xreCalA	47.2	82.7	60.5	29.8	27.4	49.4	87.5
otL10bT	81.0	49.0	55.7	13.7	18.3	72.5	15.4
BckExtA	63.7	53.9	54.2	14.7	25.2	25.9	71.9
BckBigA	63.8	53.7	52.1	14.6	25.2	25.9	71.9
rmitindA	53.5	37.9	60.1	21.5	21.2	55.7	16.5
otL10FT	32.3	49.3	69.9	14.5	24.6	64.6	15.2
xrceNoRA	39.9	63.5	45.7	18.7	24.5	39.8	45.5
BckLitA	60.9	54.1	69.9	11.1	23.3	44.7	3.2
otL10rvlT	51.3	25.2	29.7	13.0	16.1	77.8	61.5
DUTHsdtA	55.3	39.4	55.4	6.7	16.0	36.1	12.5
DUTHsdeA	55.3	39.4	55.4	6.7	16.0	36.1	12.5
DUTHlrgA	55.3	39.4	55.4	6.7	16.0	36.1	12.5
rmitmlsT	16.4	40.1	32.2	16.3	18.6	48.5	15.1
URSK70T	15.4	8.8	33.2	26.0	6.0	56.8	12.3
URSLSIT	16.0	5.1	33.2	27.3	6.0	56.8	13.1
rmitmlfT	16.4	44.8	17.3	7.1	21.0	49.7	7.9
ITD	28.0	74.7	7.2	11.1	14.8	4.3	13.2
URSK35T	13.1	9.5	26.4	6.5	3.2	51.8	9.3
tcd1	1.3	0.2	0.0	19.7	0.0	11.5	3.8
CUTACe	97.6	100.0	99.4	98.0	97.0	88.9	71.1
STEMIR	-	-	99.7	-	-	-	-
TOPICIR	-	-	-	-	-	-	-

Table 7.17 2010 Learning Task Recall Comparison at 50,000 Cutoff

	Topics						
Runs	201	202	203	204	205	206	207
xreLogA	65.0	88.2	73.3	44.2	39.3	78.2	92.2
xreCalA	65.0	88.2	73.3	44.2	39.3	78.2	92.2
rmitindA	79.3	59.4	79.9	57.2	41.8	76.7	18.5
otL10bT	80.7	58.0	84.2	26.9	38.8	99.4	15.4
otL10rvlT	69.0	52.7	56.2	40.6	28.7	82.8	76.6
otL10FT	50.7	63.3	87.4	20.7	38.3	69.6	17.3
BckExtA	69.7	67.8	60.3	41.7	37.9	37.8	73.4
BckBigA	69.8	67.6	55.8	41.7	37.9	37.8	73.4
xrceNoRA	65.4	65.7	61.5	22.1	37.8	49.0	90.8
DUTHsdtA	80.4	58.1	74.1	22.8	40.2	67.9	13.8
DUTHsdeA	80.4	58.1	74.1	22.8	40.2	67.9	13.8
DUTHlrgA	80.4	58.1	74.1	22.8	40.2	67.9	13.8
BckLitA	68.8	63.8	76.2	13.8	36.5	49.0	4.2
rmitmlsT	23.9	55.8	52.9	21.5	29.3	55.7	15.4
rmitmlfT	24.0	53.4	20.5	18.7	28.1	56.0	11.9
URSK70T	17.2	10.3	39.4	32.8	12.2	73.6	15.6
URSLSIT	18.1	14.4	39.4	28.5	12.2	73.6	14.4
URSK35T	15.1	12.0	35.9	21.0	6.3	56.1	13.6
ITD	28.8	82.2	9.6	12.1	20.1	4.4	30.5
tcd1	15.9	0.4	12.0	23.1	5.3	38.5	10.8
CUTACe	-	-	-	-	-	-	-
STEMIR	-	-	-	-	-	-	-
TOPICIR	-	-	-	-	-	-	-

Table 7.18 2010 Learning Task Recall Comparison at 100,000 Cutoff

	Topics						
Runs	201	202	203	204	205	206	207
xreLogA	94.5	91.0	82.5	69.9	51.4	78.5	93.0
xreCalA	94.5	91.0	82.5	69.9	51.4	78.5	93.0
rmitindA	88.2	68.4	95.9	74.5	58.1	80.4	18.8
otL10bT	81.0	63.7	83.2	36.0	55.5	97.7	24.8
otL10rvlT	86.7	60.5	76.2	65.5	58.4	100.6	82.7
otL10FT	74.4	73.5	93.7	37.5	61.8	70.5	18.8
BckExtA	73.4	70.1	59.9	64.8	56.8	61.2	73.1
BckBigA	73.4	70.1	60.4	64.8	56.9	61.2	73.1
xrceNoRA	64.8	70.0	66.8	31.7	49.4	72.7	85.7
DUTHsdtA	85.6	70.0	85.5	65.3	58.0	72.9	16.3
DUTHsdeA	85.6	70.0	85.5	65.3	58.0	72.9	16.3
DUTHlrgA	85.6	70.0	85.5	65.3	58.0	72.9	16.3
BckLitA	72.5	72.8	88.4	30.6	56.1	60.3	8.9
rmitmlsT	42.4	67.1	57.4	37.0	36.9	59.2	15.3
rmitmlfT	42.3	65.6	28.1	39.9	36.6	58.0	13.3
URSK70T	19.0	10.7	42.1	40.8	13.3	74.7	17.1
URSLSIT	19.8	19.4	42.1	43.6	13.3	74.7	20.9
URSK35T	16.2	13.3	37.0	33.9	13.2	76.5	16.0
ITD	34.2	86.7	11.2	18.2	29.2	8.3	53.9
tcd1	35.1	16.9	40.4	45.9	22.0	74.2	70.5
CUTACe	-	-	-	-	-	-	-
STEMIR	-	-	-	-	-	-	-
TOPICIR	-	-	-	-	-	-	-

Table 7.19 2010 Learning Task Recall Comparison at 200,000 Cutoff

	Topics						
Runs	201	202	203	204	205	206	207
xreLogA	97.1	97.8	91.3	73.8	66.7	77.8	92.0
xreCalA	97.1	97.8	91.3	73.8	66.7	77.8	92.0
rmitindA	92.3	72.5	98.0	79.2	85.0	80.9	19.8
otL10bT	88.4	67.8	84.5	49.5	65.6	98.3	51.1
otL10rvlT	88.3	64.5	83.4	85.2	82.9	99.6	86.6
otL10FT	89.2	96.6	97.7	68.8	81.1	88.4	21.7
BckExtA	74.0	75.4	71.4	67.5	75.4	85.0	80.9
BckBigA	74.1	75.4	66.4	67.4	75.4	85.0	80.9
xrceNoRA	73.5	76.1	79.7	35.2	58.7	78.9	92.0
DUTHsdtA	90.9	72.1	97.5	98.0	80.9	88.2	18.7
DUTHsdeA	90.9	72.1	97.5	98.0	80.9	88.2	18.7
DUTHlrgA	90.9	72.1	97.5	98.0	80.9	88.2	18.7
BckLitA	74.9	75.7	85.4	42.5	77.8	63.1	11.7
rmitmlsT	58.6	72.2	64.5	45.6	54.2	61.8	16.3
rmitmlfT	57.2	70.4	47.6	47.5	52.8	62.3	15.7
URSK70T	18.9	13.0	44.5	62.2	24.6	88.9	22.6
URSLSIT	21.0	21.1	44.5	50.6	24.6	88.9	22.5
URSK35T	25.1	15.2	45.1	40.5	27.7	91.8	18.3
ITD	44.8	88.3	19.5	41.6	36.1	26.4	74.7
tcd1	55.3	85.0	76.1	76.2	53.3	98.8	87.4
CUTACe	-	-	-	-	-	-	-
STEMIR	-	-	-	-	-	-	-
TOPICIR	-	-	-	-	-	-	-

7.2.2 TREC Legal Track Interactive Task

The Interactive task of the TREC legal track [22] represents the process of using humans and technology, in consultation with topic authorities, to identify as accurate as possible all relevant documents in the collection, while simultaneously minimizing the number of false positives. The purpose of the task design was to arrive at a task that modelled more completely and accurately the objectives and conditions of eDiscovery in the real life scenario. The design is meant to measure the effectiveness of various approaches to the retrieval/classification of electronic documents.

The task simulates real life litigation scenario which begins with complaints and associated with the complaints are document requests (topics) that specify the categories of documents which must be located and produced. The goal of a team participating in a given topic is to retrieve all, and only, documents relevant to that topic as defined by the topic authority. The topic authority plays the role of a senior attorney who is charged with overseeing a client's response to a request for production and who, in that capacity, must certify to the court that the client's response to the request is complete and correct. In keeping with that role, it is this attorney who, weighing considerations of genuine subject-matter relevance as well as pragmatic considerations of legal strategy and tactics, holds ultimate responsibility for deciding what is and is not to be considered responsive for the purposes of documents production.

In the Interactive task, the topic authority determines what is and is not to be considered relevant to a target topic. Each team can ask, for each topic for which it plans to submit results, for up to ten hours of a topic authority's time for purposes of clarifying a topic. A team can call upon a topic authority at any point in the exercise, from the kickoff of the task to the deadline for the submission of results. How a team makes use of the topic authority's time is largely unrestricted; a team can ask the topic authority to pass judgment on exemplar documents; a team can submit questions to the topic authority by email; a team can arrange for conference calls to discuss aspects of the topic. The assessment for responsiveness of documents was carried out by professional document review firms using dual assessment sample and finalized by the agreement of the assessors. Disagreement on any documents were adjudged.

In evaluating the effectiveness of approaches to assessing the relevance of email messages, one must decide whether one wants to assess effectiveness at the message level (i.e., treat the parent email together with all of its attachments as the unit of assessment) or to assess effectiveness at the document level (i.e., treat each of the components of an email message; the parent email and each child attachment as a distinct unit of assessment). For the 2010 interactive exercise, in an effort to gather data on both levels, the participants are required to submit their results at the document level (in order to enable document-level analysis) from which the derive message-level values are then derived, which served as the basis for scoring. The specific rules governing the assignment of assessments were outlined as follow:

- "A parent email should be deemed relevant either if in itself, it has content that meets the definition of relevance or if any of its attachments meet that definition; contextual information contained in all components of the email message should be taken into account in determining relevance."
- "An email attachment should be deemed relevant if it has content that meets the topic authority's definition of relevance; in making this determination, contextual information contained in associated documents (parent email or sibling attachments) should be taken into account."
- "A message will count as relevant if at least one of its component documents; parent email or attachments has been found relevant."
- "For purposes of scoring, the primary level is the message-level; document-level analysis is on between documents reviewed and supplementary."

7.2.2.1 Topics for the 2010 Interactive Task

The 2010 interactive task consists of four topics. We focus on three as the fourth topic which borders on defendants withholding some documents on the basis of a claim of privilege among others is out of scope of this work. Topics 301 specifies the requirements for responsive documents as follows:

"All documents or communications that describe, discuss, refer to, report on, or relate to onshore or offshore oil and gas drilling or extraction activities, whether past, present or future, actual, anticipated, possible or potential, including, but not limited to, all business and other plans relating thereto, all anticipated revenues therefrom, and all risk calculations or risk management analyses in connection therewith."

Topic 302 states the following description:

"All documents or communications that describe, discuss, refer to, report on, or relate to actual, anticipated, possible or potential responses to oil and gas spills, blowouts or releases, or pipeline eruptions, whether past, present or future, including, but not limited to,

any assessment, evaluation, remediation or repair activities, contingency plans and/or environmental disaster, recovery or clean-up efforts".

While topic 303 stipulates that:

"All documents or communications that describe, discuss, refer to, report on, or relate to activities, plans or efforts (whether past, present or future) aimed, intended or directed at lobbying public or other officials regarding any actual, pending, anticipated, possible or potential legislation, including but not limited to, activities aimed, intended or directed at influencing or affecting any actual, pending, anticipated, possible or potential rule, regulation, standard, policy, law or amendment thereto."

The estimated yield of relevant documents for each topic in the collection is presented in Table 7.20:

7.2.2.2 Results of the combined unsupervised algorithm on the 2010 interactive task

Table 7.21 shows the combined algorithm's performance (recall, precision and F1) on the three interactive task topics. For each of the topics, an optimal cutoff point has been chosen. The cutoff points for topics 301, 302 and 303 are 63,000, 580 and 12,500 respectively.

Table 7.22 and 7.23 shows the results of the stem-based search and topic-based retrieval respectively using the same cut-off point as the combined unsupervised technique both at the same cutoff points as CUTACe for the three topics.

Finally, Table 7.24 presents a juxtaposition of the combined unsupervised algorithm with task participated systems. Values for each of the metrics are estimated at 95% confidence interval for the participated runs. The best performance metrics' values for each of the topics is presented in bold.

7.3 Discussion of Results

We take a look at the performance of the combined unsupervised algorithm first with component algorithms and then with supervised systems which participated in TREC legal track tasks.

7.3.1 Discussion of Learning Task Results

The 2011 learning task shows that tradtional stem-based search and topic-based retrieval achieves better performance than the combined unsupervised algorithm. The three methods

Table 7.20 Estimated Yield of Responsive Documents for Topics in The Collection

	Relevant Messages		Of Full Collection	
Topics	Estimate	95% Confidence Interval	Estimate	95% Confidence Interval
301	18,973	(16,688, 21,258)	0.042	(0.037, 0.047)
302	575	(174, 976)	0.001	(0.0004, 0.002)
303	12,124	(11,261, 12,987)	0.027	(0.025, 0.029)

Table 7.21 Results of the combined unsupervised algorithm on the 2010 interactive task

Topics	Recall	Precision	F1
301	0.457	0.181	0.259
302	0.249	0.213	0.229
303	0.159	0.215	0.183

Table 7.22 Results of stem-based search on the 2010 interactive task

Topics	Recall	Precision	F1
301	0.424	0.154	0.225
302	0.123	0.161	0.139
303	0.156	0.212	0.180

Table 7.23 Results of topic-based retrieval on the 2010 interactive task

Topics	Recall	Precision	F1
301	0.407	0.142	0.211
302	0.064	0.055	0.059
303	0.153	0.187	0.169

Table 7.24 Performance Comparison Table for the 2010 interactive task

Topics	Run	Recall	Precision	F1
301	CS	0.165	0.579	0.256
	IT	0.205	0.295	0.242
	SF	0.239	0.193	0.214
	IS	0.027	0.867	0.052
	UW	0.019	0.578	0.036
	CUTACe	0.457	0.181	0.259
	STEMIR	0.424	0.154	0.225
	TOPICIR301	0.407	0.142	0.211
302	UM	0.200	0.450	0.277
	UW	0.169	0.732	0.275
	MM	0.115	0.410	0.180
	LA	0.096	0.481	0.160
	IS	0.090	0.693	0.160
	IN	0.135	0.017	0.031
	CUTACe	0.249	0.213	0.229
	STEMIR	0.123	0.161	0.139
	TOPICIR	0.064	0.055	0.059
303	EQ	0.801	0.577	0.671
	CB2	0.572	0.705	0.631
	CB1	0.452	0.734	0.559
	UB	0.723	0.300	0.424
	IT	0.248	0.259	0.254
	UW	0.134	0.773	0.228
	CUTACe	0.159	0.215	0.183
	STEMIR	0.156	0.212	0.180
	TOPICIR	0.153	0.187	0.169

produce results up till 20,000 cutoff point. However, comparing the results with other runs that participated in the task, topic-based retrieval outperforms every other runs in recall and F1 at 2,000 cutoff point, stem-based search has the best recall at 20,000 cutoff point.

Although we do not know the logic of scoring used by the evaluation tool but we suspect the performance behaviour of the three methods which shows stem-based search and topic-based retrieval outperforming the combined unsupervised method might be connected to variance in the number of matching documents produced for each of the methods and ranked accordingly according to their VSM similarity score and the probability estimate computed based on the rank. The number of matching documents for topics 401, 402 and 403 using the stem-based search are 541,899, 215,922 and 155,236 respectively. The topic-based retrieval matches 457,423, 67,262 and 44,186 documents for topics 401, 402 and 403 respectively while the combined algorithm matches 555,772, 554,901 and 555,740 documents for topics 401, 402 and 403 respectively. According to the learning task guidelines, all documents are supposed to be ranked based on the probability estimate.

For the 2010 learning task, the stem-based and topic-based retrieval do not produce results at any of the designated cutoff point except at 20,000 cutoff point where the stem-based search produces result. Careful study reveals this is as a result of the short nature of responsiveness descriptions used for the topics. The combined algorithm on the hand, produces result for all the topics at the 20,000 cutoff point only as a result of the query expansion on the initial query derived from the topics. At the 20,000 cutoff point, the combined technique achieves recall of 97.6, 100, 99.4, 98.0, 97.0, 88.9, and 71.1 respectively, the highest across the topics except topic 207.

7.3.2 Discussion of Interactive Task Results

The interactive task scores system outputs based on the number of responsive documents retrieved at a chosen cutoff point, in contrast to the learning task which uses ranking and probability estimates. As mentioned earlier, this is an appropriate model of real eDiscovery scenario. Evaluation using the three methods on topics 301, 302 and 303 shows the combined unsupervised approach consistently outperforming the other two methods. It is also noeworthy that apart from the chosenl cutoff points, several other cutoff point considered still shows the performance of the combined approach better than the other two.

Comparing the results with other runs that participated in the exercise, the combined approach performs better in topic 301 with 0.259 F1 , third in topic 302 and the last in topic 303. Furthermore, a fairer comparison would have been possible had all the runs participated in the three topics. This is difficult due to the selective participation of runs. Only UW participated in the three topics, performing far below CUTACe in topic 301 and better in topics 302 and 303. IT participates only in topics 301 and 303, achieving better performance than CUTACe in topic 303 and below CUTACe in topic 301. IS participated in topics 301 and 302 with performance below CUTACe in both.

Chapter 8

Related Work

8.1 Related Work on Sense-Based Information Retrieval

A considerable amont of previous works have been carried out on word sense based information retrieval (IR) with reports of conflicting performances. One of the earliest works on sense-based retrieval is the work of Weiss [106]. He used a disambiguator to resolve the senses of five ambiguous hand picked words in the ADI collection. He reported a 1% improvement in IR performance.

Wallis [104] used a disambiguator in an elaborate experiment which replaced the words in a text collection by the text of their dictionary definitions. When replacing a word by its definition, a disambiguator was used to select the definition that most represented the word. Wallis performed tests on the CACM and TIME collections with no significant improvement in IR performance.

Krovetz and Croft [52] studies sense matches between terms in queries and document collection. Their conclusion was that the gains of word sense disambiguation (WSD) in information retrieval are below expectation because query words have skewed sense frequency distributions and the collocation effect from other query terms already performs some sense disambiguation.

Sanderson [93, 91] used pseudowords to introduce artificial word ambiguity in order to study the impact of sense ambiguity on information retrieval. He concluded that high sense disambiguation accuracy is an essential requirement for word sense disambiguation to be of any use to an IR system and that word sense ambiguity is only problematic to an IR system when it is retrieving for very short queries.

Gonzalo et al. [35] converted a manually sense annotated corpus, SemCor to an IR collection to examine the gains of retrieving from an accurately disambiguated document collection against incorrect disambiguation on IR. They obtained significant improvements by representing documents and queries with accurate senses as well as synsets. Results from their experiment shows that a disambiguation error of 30% is just higher than the baseline 54.4% and an error rate of 60% is just below 49.1%. Although an error degree similar to the baseline was not provided but there was a speculation that such a degree lies somewhere between 40-50% and that this can still improve IR performance. In a later work [34], they concluded that part-of-speech (POS) information is discriminative for IR purposes.

Several works attempted to disambiguate terms in both queries and documents with the senses obtained from supervised word sense disambiguation, and then uses the senses to perform indexing and retrieval. Voorhees [99] used the hypernymy (is-a) and hyponymy (instance-of) relations in WordNet [70, 31] to disambiguate polysemous noun synsets in a text. She ranked the senses based on the amount of co-occurrence between the word's context and words in the hood of its synsets. In her experiment, the performance of sense-based retrieval is worse than stem-based retrieval on all test collections. Her analysis showed that inaccurate WSD caused the poor results.

Stokoe and Tait [96] did an experimental comparison of term frequency versus word sense frequency on web search using a statistical system trained using the Brown1 part of Semcor1.6 which is distributed with WordNet. They concluded that the overall results was disappointing though their system was able to improve precision by 0.0003%, which is statistically insignificant. They gave possible explanations for the poor results to the weak topic distillation strategy used in their system and inadequate training data due to high WordNet frequency statistics for words that had not been encountered in their training system.

Stokoe et al. [95] in another work, employed a fine-grained WSD system to disambiguate terms in both the text collections and queries in an experiment. Their evaluation on TREC collections using word sense based vector space model achieved significant improvements over a standard term based vector space model.

Kim et al. [51] shifted away from the use of fine-grained sense inventory, and instead tagged words with 25 root senses of nouns in WordNet. Their retrieval method maintained the stem-based index and adjusted the term weight in a document according to its sense matching with tagged senses in the query. They attributed the improvement achieved on TREC collections to their coarse-grained, consistent, and flexible sense tagging method. The integration of senses into the traditional stem-based index overcomes some of the negative impact of disambiguation errors.

In contrast to the use of predefined sense inventories, Schütze and Pedersen [94] induced sense inventory directly from a text retrieval collection. For each word, its occurrences were clustered into senses based on the similarities of their contexts. Their experiments showed that using senses improved retrieval performance, and the combination of word-based ranking and sense-based ranking can further improve performance. However, the clustering process of each word is a time consuming task. Because the sense inventory is collection dependent, it is also hard to expand the text collection without repeating preprocessing.

Zhong and Ng [112] experimented on short queries. They incorporated word senses into a language modelling technique and integrated synonym relations using stem-based term expansion on both queries and documents. In their approach, they carried out WSD in a supervised setting for the query expansion. Their conclusion was that there is a significant improvement over state-of-the-art IR system when evaluated on a TREC task.

Furthermore, some studies investigated the effects of query expansion by using semantic knowledge from WordNet for query and/or document expansion which reported improvements improvements in IR systems [15, 1, 30, 62, 63, 100]. Otegi et al. [76] used a random walks over a graph representation of WordNet [70, 31] to obtain semantic related terms in both queries and documents for expansion. They concluded that their experiments shows improvement over a language modelling baseline while complementary with pseudo-relevance feedback (PRF).

A general conclusion that can be deduced from various attempts at sense-based information retrieval is that high accuracy in determination of appropriate word senses in queries and documents is a sine qua non if any improvement in the performance of information retrieval must be achieved.

8.2 Related Work on LDA for Information Retrieval

Topic modelling in general is aged in its application to information retrieval. Latent dirichlet allocation (LDA) is one of the most popular and currently has several applications in information retrieval. One of the earliest work which attempt to apply latent dirichlet allocation to information retrieval is the work of Wei and Croft [105]. They use the query likelihood of a document model generating a query for ad hoc retrieval. They used Gibbs sampling for inference approximation in LDA. In their evaluation, they compared their approach with cluster-based approach. The following conclusions were made:

- LDA based model consistently outperforms the cluster-based approach
- Estimation of LDA model on IR task is feasible with suitable parameters.

Lu et al. [64] carried out an empirical study on the performance of pLSI and LDA on three text mining applications; document clustering, text categorization and ad hoc retrieval. They concluded with the following observations:

- When topic model is used inappropriately, it could hurt performance. They suggested combination of low dimensional representation of topic models with the original high dimensional representation tends to produce most robust and effective results.
- The performance of LDA on all tasks is quite sensitive to the setting of its hyperparameter, and the optimal setting varies according to how the model is used in a task and that the choice of this parameter is dependent on the collection.
- The problem of getting to only local maxima does not necessarily affect task performance much in which there is even no guarantee of getting to it.
- pLSI and LDA perform similarly on document clustering, where the most significant topic is used as the cluster label and ad hoc retrieval, where only the topic-word distributions are different for the two models but LDA works better in text categorization where the topic-document distribution is used as a low dimension representation.
- High computational complexity is a major limitation for the two models.

Park and Ramamohanarao [79] augmented standard keyword search by automatically learning topics from a corpus using LDA which are presented to user(s). User relevance feedback from the choice of topics presented is then used to reformulate the initial query through query expansion. Experimental results of their work shows this user feedback mechanism can improve performance.

Deveaud et al [26] experimented with a topic model-based feedback approach by learning LDA topics from top ranked documents using ad hoc retrieval measuring the semantic coherence of the learned topics. They concluded that estimating feedback query model using topics greatly enhances ad hoc information retrieval.

8.3 Beyond Bag of Words

Ad hoc information retrieval systems are used to represent documents as bag of words using vector space representation which takes advantage of their frequency to compute their similarity to queries. On the other hand, word sense based IR systems are used to give semantic representation of documents based on the meaning in order to compute their similarity with queries.

In this work, we propose and implement an hypothesis beyond the bag of words principle and semantic representation of documents by taking advantage of topicality of documents in a collection in conjuction with the semantics of the terms in topics, thereby presenting a vector space model of semantic terms; that is from frequency to meaning and from bag of words to topicality, we have been able to model documents based on these assumptions bringing together two different fields of topic modelling and natural language processing.

Chapter 9

Conclusion and Future Work

9.1 Conclusion

The retrieval of relevant documents in response to a production request from an opposing party in litigations is a major problem facing lawyers and litigants. Present state-of-the-art eDiscovery search algorithms and techniques focus on supervised approaches which are project-based. This involves using the specific nature of thedocument collection and case at hand to develop search systems for retrieving relevant documents. Specifically, they are trained on manually labelled examples. These approaches, were to some extent, able to satisfy the needs of clients because of their superior performance compared to unsupervised approaches. However, due to huge cost expended on this process including developing training datasets and the systems, training and staffing among other miscellaneous expenses, coupled with the inability for reuse of the end systems, organisations are trying to cut cost by trying to seek solutions for general approach for retrieval of relevant documents which allows reusability across collections and cases. In fact, due to this huge financial implications, several good cases have been abandoned or settled out of court. There is need for more effort in the direction of unsupervised approaches to search in eDiscovery.

In this thesis, a combined unsupervised ad hoc approach for retrieval and classification in eDiscovery is explored. This approach provides a general case tailored towards any collection. We have employed semantic aproach together with topic mining in an automated fashion. The technique uses a hybrid word sense disambiguation which heuristically resolves conflict in two knowledge-based word sense disambiguation algorithms; the modified Lesk algorithm and the Jiang & Conrath similarity measure. Latent dirichlet allocation (LDA) model with Gibbs sampling is employed for mining topical relations from the documents after which words in the mined topics are disambiguated using the hybrid word sense disambiguaton algorithm. This enhances the efficiency of the techniques due to the computational complexities associated with word sense disambiguation especially when applied to huge collections. This answers the second research questions about efficiency of sense-based approaches to /searchclassification in eDiscovery.

Empirical results obtained on TREC legal tasks data shows this technique outperforms stem-based retrieval and topic-based retrieval and comparable with supervised techniques in the interactive task, which in our opinion, is the most appropriate for evaluating this type of ad hoc search. This answers the first research question about whether a semantic, ad hoc, automated unsupervised technique can improve retrieval effectiveness.

Furthermore, most state-of-the-art, intelligent unsupervised systems have associated problems of scalability and semantics; implementations issues due to huge size of collections, polysemy and synonymy issues. We have been able to provide answers to these issues by a novel approach using a combined topic modelling and ambiguity resolution techniques

9.2 Future Work

Apart from the latent dirichlet allocation (LDA) model being able to group collocated words and using word sense disambiguation WSD) on the topic terms to determine their contexualt meaning, there still exist other relationships among topic words and other world objects which are not fully captured by the disambiguated terms. In the future, we recommend the exploration of these relationships which exist among extracted topic terms in documents and queries for the purpose of carrying out common sense reasoning on them which will provide more insight into the search for relevance in collections. Furthermore, the work can be extended to web search engines majority of whose basic technologies are still based keyword search.

Most importantly, careful study of the results shows that word sense disambiguation improves traditional stem-based IR effective performance; recall and precision, however, WSD is very expensive in computational time, hence the need to model documents as topics. We recommend for further research, exploration of efficient algorithms that can disambiguate the original collection in reasonable time for term expansion. This will enhance the effective performance of sense-based retrieval and make it practicable in real life scenario.

References

[1] Agirre, E., Arregi, X., and Otegi, A. (2010). Document expansion based on wordnet for robust ir. In *Proceedings of the 23rd international conference on computational linguistics*, pages 9–17. Association for Computational Linguistics Stroudsburg, PA, USA.

[2] Agirre, E. and Rigau, G. (1996). Word sense disambiguation using conceptual density. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 16–22. Association for Computational Linguistics Stroudsburg, PA, USA.

[3] Amati, G. (2003). Probabilistic models for information retrieval based on divergence from randomness. *PhD Thesis.*

[4] Anaya-Sánchez, H., Pons-Porrata, A., and Berlanga-Llavori, R. (2007). Tkb-uo: Using sense clustering for word sense disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 322 –325. Association for Computational Linguistics Stroudsburg, PA, USA.

[5] Apache (© 2011-2012). *Apache Lucene*. https://lucene.apache.org/.

[6] Aslam, J. A. and Yilmaz, E. (2007). Inferring document relevance from incomplete information. In *Proceedings of the 16th ACM conference on Conference on information and knowledge management*, pages 633–642. ACM Press.

[7] Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley.

[8] Banerjee, S. and Pedersen, T. (2002). An adapted lesk algorithm for word sense disambiguation using wordnet. In *Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing*, pages 136–145. Springer-Verlag London, UK.

[9] Blei, D. M., Griffiths, T. L., Jordan, M. I., and Tenenbaum, J. B. (2004). Hierarchical topic models and the nested chinese restaurant process. *Advances in Neural information processing systems 16*.

[10] Blei, D. M. and Jordan, M. (2002). Modeling annotated data. *Technical Report*, UCB//CSD-02-1202.

[11] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

[12] Buckley, C., Salton, G., Alan, J., and Singhal, A. (1995). Automatic query expansion using smart: Trec3. In *In Proceedings of the 3rd Conference on Text Retrieval (TREC-3)*, pages 69–80. TREC.

[13] Buckley, C. and Voorhees, E. M. (2000). Evaluating evaluation measure stability. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–40. ACM Press.

[14] Budanitsky, A. and Hirst, G. (2006). Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

[15] Cao, G., Nie, J.-Y., and Bai, J. (2005). Integrating word relationships into language models. pages 298–305. ACM New York, NY, USA.

[16] Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 44(1):1557–7341.

[17] Carpineto, C., Romano, G., and Giannini, V. (2002). Improving retrieval feedback with multiple term-ranking function combination. *ACM Transactions on Information Systems (TOIS)*, 20(3):259–290.

[18] Chen, P., Ding, W., Bowes, C., and Brown, D. (2009). A fully unsupervised word sense disambiguation method using dependency knowledge. pages 28 –36. Association for Computational Linguistics Stroudsburg, PA, USA.

[19] Cleverdon, C. W. (1970). The effect of variations in relevance assessments in comparative experimental tests of index languages. *Cranfield Library Report*, No. 3.

[20] Cleverdon, C. W. and Keen, M. (1966). *Factors Affecting the Performance of Indexing Systems*. Vol 2. ASLIB, Cranfield Research Project. Bedford, UK.

[21] Conrad, J. G. (2010). E-discovery revisited: the need for artificial intelligence beyond information retrieval. *Artificial Intelligence and Law*, 18(4):321–345.

[22] Cormack, G. V., Grossman, M. R., Hedin, B., and Oard, D. W. (2010). Overview of the trec 2010 legal track. In *Proceedings of 19th Text REtrieval Conference*. TREC.

[23] Counsel, C. (2006). The American Bar Association (ABA), section of litigation, committee on Corporate Counsel, http://www.abanet.org/litigation/committees/corporate/.

[24] Cowie, J., Guthrie, J., and Guthrie, L. (1992). Lexical disambiguation using simulated annealing. In *Proceedings of the 14th conference on Computational linguistics(COLING)*, pages 359–365. Association for Computational Linguistics, Stroudsburg, PA, USA.

[25] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society For Information Science*, 41(6):391–407.

[26] Deveaud, R., SanJuan, E., and Bellot, P. (2013). Are semantically coherent topic models useful for ad hoc information retrieval? In *Proceedings of 51st Annual Meeting of the Association of computational Linguistics*. Association for Computational Linguistics Stroudsburg, PA, USA.

[27] EDRM. *Electronic Discovery Reference Model*. http://www.edrm.net/resources/edrm-stages-explained.

[28] EDRM. *The Metrics Model*. http://www.edrm.net/projects/metrics/metrics-model.

[29] Efthimiadis, E. N. (1996). Query expansion. In *Annual Review of Information Systems and Technology (ARIST)*, pages 121–187. M. E. Williams, Ed. ASIS&T.

[30] Fang, H. (2008). A re-examination of query expansion using lexical resources. In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics: Human Language Technologies*, page 139–147. Association for Computational Linguistics Stroudsburg, PA, USA.

[31] Fellbaum, C. (1998, ed). *An Electronic Lexical Database*. MIT Press.

[32] Furnas, G. W., Landauer, T. K., Gomez, L., and Dumais, S. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971.

[33] Gilks, W., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. Chapman & Hall, Suffolk.

[34] Gonzalo, J., Penas, A., and Verdejo, F. (1999). Lexical ambiguity and information retrieval revisited. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 195 –202.

[35] Gonzalo, J., Verdejo, F., Chugur, I., and Cigarran, J. (1998). Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING-ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 38–44.

[36] Griffiths, T. L. and Steyvers, M. (2002). A probabilistic approach to semantic representation. In *In Proceedings of the 24th Annual Conference of the Cognitive Science Society*.

[37] Griffiths, T. L. and Steyvers, M. (2003). Prediction and semantic association. *Neural information processing systems 15*.

[38] Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. In *Proceedings of the National Academy of Sciences*, pages 5228–5235. National Academy of Sciences.

[39] Grossman, M. R. and Cormack, G. V. (2011). Technology-assisted review in e-discovery can be more effective and more efficient than exhaustive manual review. *Richmond Journal of Law and Technology.*, XVII, Issue 3:1–48.

[40] Grossman, M. R., Cormack, G. V., Hedin, B., and Oard, D. W. (2011). Overview of the trec 2011 legal track. In *Proceedings of 20th Text REtrieval Conference*. TREC.

[41] Harman, D. (1992). *Relevance feedback and other query modification techniques*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[42] Harris, Z. (1954). Distributional structure. *Word*, 10(23):146–162.

[43] Hirst, G. and St-Onge, D. (1998). *Lexical chains as representations of context in the detection and correction of malaproprisms*.

[44] Hoffman, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM New York, NY, USA.

[45] Ji, H. (2010). One sense per context cluster: Improving word sense disambiguation using web-scale phrase clustering. In *Proceedings of the 4th Universal Communication Symposium (IUCS)*, pages 181–184. IEEE.

[46] Jiang, J. J. and Conrath, D. W. (1999). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th International Conference on Research in Computational Linguistics*, pages 19–33.

[47] Jones, K. S. and Bates, R. G. (1977). Report on the need for and the provision of an 'ideal' information retrieval test collection. *British Library Research and Development Report*, No. 5428.

[48] Jones, K. S. and van Rijsbergen, C. J. (1975). Report on the need for and the provision of an 'ideal' information retrieval test collection. *British Library Research and Development Report*, No. 5266:43.

[49] Jones, K. S., Walker, S., and Robertson, S. E. (2000). A probabilistic model of information retrieval: Development and comparative experiments. *IP&M*, 36(6):779–808.

[50] K. Toutanova, D. Klein, C. M. and Singer., Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180. Association for Computational Linguistics Stroudsburg, PA, USA.

[51] Kim, S. B., Seo, H. C., and Rim, H. C. (2004). Information retrieval using word senses: root sense tagging approach. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 258–265. ACM New York, NY, USA.

[52] Krovetz, R. and Croft, W. B. (1992). Lexical ambiguity and information retrieval. *ACM Transactions on Information Systems (TOIS)*, 10(2):115– –141.

[53] Kurland, O., Lee, L., and Domshlak, C. (2005). Better than the real thing?: iterative pseudoquery processing using cluster-based language models. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 19–26. ACM Press.

[54] Lafferty, J. and Zhai, C. (2001). Document language models, query models and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieva*, pages 111–119. ACM New York, NY, USA.

[55] Lang, K. (1995). Newsweeder: Learning to filter netnews. In *Proceedings of 12th International Conference on Machine Learning*.

[56] Leacock, C., Miller, G. A., and Chodorow, M. (1998). Using corpus statistics and wordnet relations for sense identification. *Computational Linguistics - Special issue on word sense disambiguation*, 24(1):147–165.

[57] Lee, K. S., Croft, W., and Allan, J. (2008). A cluster-based resampling method for pseudorelevance feedback. In *Proceedings of the 31th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 235–242. ACM Press.

[58] Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th ACM-SIGDOC Conference*, pages 24–26. ACM New York, NY, USA.

[59] Lewis, D., Agam, G., Argamon, S., Frieder, O., Grossman, D., and Heard, J. (2006). Building a test collection for complex document information processing. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 665–666. ACM Press.

[60] Lewis, D. D., Yiming, Y., Tony, R., and Fan, L. (2004). Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5(2004):361–397.

[61] Lin, D. (1998). An information theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

[62] Liu, S., Liu, F., Yu, C., and Meng, W. (2004). An effective approach to document retrieval via utilizing wordnet and recognizing phrases. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 266–272. ACM New York, NY, USA.

[63] Liu, S., Yu, C., and Meng, W. (2005). Word sense disambiguation in queries. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 525–532. ACM New York, NY, USA.

[64] Lu, Y., Mei, Q., and Zhai, C. (2010). Investigating task performance of probabilistic topic models: an empirical study of plsa and lda. *Information Retrieval*, 14(2):178–203.

[65] Magalhães, J. M. C. (2008). *Statistical Models for Semantic-Multimedia Information Retrieval*. PhD Thesis, Imperial College of Science, Technology and Medicine Department of Computing.

[66] Manning, C. D., Raghavan, P., and Schütze, H. (1999). *Introduction to Information Retrieval*. University of Cambridge Press, Cambridge, England.

[67] McCallum, A. K. (2002). *MALLET: A Machine Learning for Language Toolkit*. http://mallet.cs.umass.edu.

[68] McCandless, M., Hatcher, E., and Gospodnetic, O. (2010). *Lucene in Action*. Manning Publications, Stamford, USA.

[69] Mihalcea, R. and Moldovan, D. I. (1999). A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 152–158. Association for Computational Linguistics Stroudsburg, PA, USA.

[70] Miller, G. A. (1995). Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41.

[71] Mitra, M., Singhal, A., , and Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–214. ACM Press.

[72] Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 30–35. Association for Computational Linguistics Stroudsburg, PA, USA.

[73] Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193(2012):217–250.

[74] Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods. In *Technical Report CRG-TR-93-1*. Department of Computer Science, University of Toronto.

[75] Oard, D. W. and Webber, W. (2013). *Information Retrieval for E-Discovery*, volume 7(2-3). Now Publishers, Delft, Netherlands.

[76] Otegi, A., Arregi, X., Ansa, O., and Agirre, E. (2014). Using knowledge-based relatedness for information retrieval. *Knowledge Information Systems*, 44(2015).

[77] Pace, N. M. and Zakaras, L. (2012). Where the money goes: Understanding litigant expenditures for producing electronic discovery. *RAND*.

[78] Pan, B., Lillian, L., and Shivakumar, V. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 79–86. Association for Computational Linguistics Stroudsburg, PA, USA.

[79] Park, L. A. F. and Ramamohanarao, K. (2009). The sensitivity of latent dirichlet allocation for information retrieval. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 176 – 188. Springer-Verlag Berlin, Heidelberg.

[80] Patwardhan, S., Banerjee, S., and Pedersen, T. (2003). Using measures of semantic re-latedness for word sense disambiguation. In *Proceedings of the 4th International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257. Springer-Verlag Berlin, Heidelberg.

[81] Ponzeto, S. P. and Navigli, R. (2010). Knowledge-rich word sense disambiguation rivaling supervised systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1522–1531. Association for Computational Linguistics Stroudsburg, PA, USA.

[82] Popescul, A., Ungar, L. H., Pennock, D. M., and Lawrence, S. (2001). Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Proceedings of 17th Conference in Uncertainty in Artificial Intelligence (UAI)*, pages 437–444. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

[83] Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137.

[84] Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1):17–30.

[85] Reisnik, P. (1995). Using information content to evaluate semantic similarity. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 448–453. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

[86] Rocchio, J. J. (1971). Relevance feedback in information retrieval. In *G. Salton(ed.): The SMART Retrieval System: Experiments in Automatic Text Retrieval*, pages 313–323. Prentice-Hall.

[87] Ruthven, I. (2003). Re-examining the potential effectiveness of interactive query expansion. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 213–220. ACM Press.

[88] Salton, G. and Lesk, M. E. (1968). Computer evaluation of indexing and text processing. *Journal of the ACM (JACM)*, 15(1):8–36.

[89] Salton, G. and McGill, M. J. (1986). *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

[90] Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

[91] Sanderson, M. (2000). Retrieving with good sense. *Information Retrieval*, 2(1):49–69.

[92] Sanderson, M. (2010a). Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, Vol. 4, No. 4 (2010):247–375.

[93] Sanderson, M. (2010b). Word sense disambiguation and information retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 142–151. Springer-Verlag New York, Inc. New York, NY, USA.

[94] Schütze, H. and Pedersen, J. O. (1995). Information retrieval based on word senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175.

[95] Stokoe, C., Oakes, M. P., and Tait, J. (2003). Word sense disambiguation in information retrieval revisited. In *Proceedings of The 12th Text REtrieval Conference (TREC 2003)*, pages 159–166. TREC.

[96] Stokoe, C. and Tait, J. (2003). Towards a sense based document representation for internet information retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 791–795. ACM New York, NY, USA.

[97] Tanner, M. (1996). *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*. Springer-Verlag, New York.

[98] Thomlinson, S. (2011). Learning task experiments in the trec 2010 legal track. In *Proceedings of 2010 Text Retrieval Conference*. TREC.

[99] Voorhees, E. M. (1993). Using wordnet to disambiguate word senses for text retrieval. In *Proceedings of the 16th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–180. ACM New York, NY, USA.

[100] Voorhees, E. M. (1994). Query expansion using lexical semantic relations. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69. Springer-Verlag New York, Inc. New York, NY, USA.

[101] Voorhees, E. M. (1998). Variations in relevance judgments and the measurement of retrieval effectiveness. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–323. ACM Press.

[102] Voorhees, E. M. and Harman, D. (2000). Overview of the sixth text retrieval conference (trec-6). *Information Processing and Management: an International Journal - The sixth text REtrieval conference (TREC-6)*, 36(1):3 – 35.

[103] Vélez, B., Weiss, R., Sheldon, M., and Gifford, D. (1997). Fast and effective query refinement. In *Proceedings of the 20th Annual International ACM Conference on Research and Development in Information Retrieval*, pages 6–15. ACM New York, NY, USA.

[104] Wallis, P. (1993). Information retrieval based on paraphrase. In *Proceedings of PACLING Conference*.

[105] Wei, X. and Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185. ACM New York, NY, USA.

[106] Weiss, S. (1973). Learning to disambiguate. *Information Storage and Retrieval*, 9:33–41.

[107] Wu., Z. and Palmer, M. (1994). Verb semantics and lexical selection. In *Proceedings of 32nd Annual Meeting of the Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics, Stroudsburg, PA, USA.

[108] Xu, J. and Croft, W. B. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM New York, NY, USA.

[109] Xu, J. and Croft, W. B. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems (TOIS)*, 18(1):79–112.

[110] Yilmaz, E. and Aslam, J. A. (2006). Estimating average precision with incomplete and imperfect judgments. In *Proceedings of the 15th ACM conference on Conference on information and knowledge management*, pages 102–111. ACM Press.

[111] Zhang, J. and Landers, G. (2015). Magic quadrant for e-discovery software. *Gartner E-Discovery Analysis Report*, G00267058.

[112] Zhong, Z. and Ng, H. T. (2010). Word sense disambiguation improves information retrieval. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 273–282. Association for Computational Linguistics Stroudsburg, PA, USA.

Appendix A

Acronyms

- **TAR** - Technology Assisted Review
- **WSD** - Word Sense Disambiguation
- **IR** - Information Retrieval
- **NLP** - Natural Language Processing
- **CUTACE** - Combined Unsupervised Technique for Automatic Classification E-Discovery
- **LDA** - Latent Dircihlet Allocation Model
- **pLSI** - Probabilistic Latent Semantic Indexing
- **LSI** - Latent Semantic Indexing
- **FRCP** - Federal Rule of Civil Procedure
- **E-Discovery/EDD** - Electronic Discovery/Electronic Data Discovery
- **TREC** - Text Retrieval Conference
- **IC** - Information Content
- **LCS** - Lowest Common Subsumer
- **TP** - True Posititve
- **TN** - True Negative
- **FP** - False Positive

- **FN** - False Negative
- **AQE** - Automatic Query Expansion
- **QE** - Query Expansion
- **IQE** - Interactive Query Expansion
- **RF** - Relevance Feedback
- **SRC** - Search Result Clustering