

TEXT REPRESENTATION USING CANONICAL DATA MODEL



HIBA JASIM HADI

UUM
Universiti Utara Malaysia

MASTER OF INFORMATION TECHNOLOGY

COLLEGE OF ARTS AND SCIENCES

UNIVERSITI UTARA MALAYSIA

2016

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission.

It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences

UUM College of Arts and Sciences

Universiti Utara Malaysia

06010 UUM Sintok

ABSTRAK

Pembangunan teknologi digital dan World Wide Web telah membawa kepada peningkatan dokumentasi-dokumentasi digital yang digunakan untuk pelbagai keperluan contohnya dalam bidang penerbitan yang telah menunjukkan perkaitan dalam meningkatkan kesedaran tentang keperluan teknik yang berkesan yang membantu dalam pencarian dan mendapatkan teks. Persembahan teks memainkan peranan yang amat penting dalam menyampaikan maksud teks dengan lebih bermakna atau tepat. Ketepatan penyampaian sesuatu teks amat bergantung kepada pemilihan kaedah teks itu dipersembahkan. Kaedah tradisional di dalam persembahan teks berdasarkan model dokumen seperti term-frequency invers document frequency (TF-IDF) tidak menitikberatkan hubungan dan makna perkataan di dalam sesuatu dokumen. Oleh itu, masalah sparsiti dan semantik yang merupakan masalah yang dominan di dalam dokumen teks masih belum menemui penyelesaian. Kajian ini mencadangkan bagaimana masalah sparsiti dan semantic dikurangkan dengan penggunaan Canonical Data Model (CDM) untuk menyampaikan teks. CDM distruktur melalui pengumpulan analisis semantik dan sintaksis. 20 kumpulan dataset berita telah digunakan untuk menguji kesahihan CDM dalam penyampaian teks dalam kajian ini. Dokumen-dokumen teks akan melalui beberapa proses pra-pemprosesan dan menghuraikan sintaksis untuk mengenal pasti struktur ayat. Dokumen teks akan melalui beberapa langkah pra-pemprosesan dan menghuraikan sintaksis untuk mengenal pasti struktur ayat dan maka kaedah TF-IDF digunakan untuk mewakili teks yang melalui CDM. Ini membuktikan bahawa CDM tepat untuk mewakili teks, berdasarkan pengesahan model melalui kajian bahasa pakar-pakar berdasarkan peratusan kaedah pengukuran persamaan.

KATA KUNCI: Perwakilan Teks, TF-IDF, CDM

ABSTRACT

Developing digital technology and the World Wide Web has led to the increase of digital documents that are used for various purposes such as publishing, in turn, appears to be connected to raise the awareness for the requirement of effective techniques that can help during the search and retrieval of text. Text representation plays a crucial role in representing text in a meaningful way. The clarity of representation depends tightly on the selection of the text representation methods. Traditional methods of text representation model documents such as term-frequency invers document frequency (TF-IDF) ignores the relationship and meanings of words in documents. As a result the sparsity and semantic problem that is predominant in textual document are not resolved. In this research, the problem of sparsity and semantic is reduced by proposing Canonical Data Model (CDM) for text representation. CDM is constructed through an accumulation of syntactic and semantic analysis. A number of 20 news group dataset were used in this research to test CDM validity for text representation. The text documents goes through a number of pre-processing process and syntactic parsing in order to identify the sentence structure. Text documents goes through a number of pre-processing steps and syntactic parsing in order to identify the sentence structure and then TF-IDF method is used to represent the text through CDM. The findings proved that CDM was efficient to represent text, based on the model validation through language experts' review and the percentage of the similarity measurement methods.

Keywords: Text Representation, TF-IDF, CDM

Acknowledgments

“Alhamdulillah’, praise be to Allah, The Most Beneficent, The Most Merciful”

It gives me great pleasure to thank and acknowledge all those who have contributed to my education journey whose results have flourished into this thesis.

First and foremost: I would like to extend my gratitude to my supervisors Dr. Azizah Bt Haji Ahmad and DR. Siti Sakira Kamaruddin for their unreserved guidance and counsel rendered from the very beginning to the completion of the research. I appreciate their kindness and support which have manifested in various ways. Without their support, guidance, and help this research would not have been successfully materialized. I cannot fully express my gratitude to the exceptional advice every time I seek enlightenment, the sharing of their knowledge from both theoretical and practical aspects, and their logical way of thinking have provided a good basis for this work. They guided me and polish my ideas, and translate them into workable solutions.

I owe my most heartiest gratitude to my wonderful parents for their patience, understanding, encourage, unlimited support, and prayers for smoothness and blessed journey of my Master, without them, I could not pursue and accomplish this task so, thank you very much.

My sincere appreciation and gratitude to the most important person who have strongly contributed, in various ways, by providing the help and support needed my dearest and nearest friend Atheer Flayh Hassan.

Ahba Fasih Hadi

Table of Contents

CHAPTER ONE INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	4
1.3 Research Questions	5
1.4 Research Objectives	6
1.5 Research Scope	6
1.6 Significance of the Research	7
CHAPTER TWO LITERATURE REVIEW	8
2.1 Text Mining.....	8
2.2 Text Classification	9
2.2.1 Supervised Classification Algorithms.....	10
2.2.2 Unsupervised Classification Algorithms	10
2.3 Text Preprocessing	11
2.3.1 Document Triage	12
2.3.2 Text Segmentation	12
2.4 Text Representation Methods	14
2.4.1 Term Based Representation	15
2.4.2 Graph Based Representation.....	28
2.5 Canonical Data Model.....	40

2.6 Summary	45
CHAPTER THREE RESEARCH METHODOLOGY	46
3.1 Introduction	46
3.2 Theoretical Study	47
3.3 Research Design.....	47
3.3.1 Document Collection	49
3.3.2 Document Pre-processing	49
3.3.3 Constructing Canonical Data Model.....	54
3.4 Evaluation	60
3.5 Summary	62
CHAPTER FOUR CONSTRUCTING CANONICAL DATA MODEL (CDM)63	
4.1 Introduction.....	63
4.2 Data Set.....	63
4.3 Implementation Tool.....	64
4.4 Main Implementation Steps	65
4.5 Document Pre-Processing	67
4.5.1 Splitting.....	67
4.5.2 Tokenization	68
4.5.3 Remove Stop Words	69
4.5.4 Stemming	70
4.5.5 Part of speech tagging (post).....	71

4.6 Constructing a CDM	73
4.6.1 Generate popularity Table by using TF-IDF:	73
4.6.2 Classes Constructing:	77
4.6.3 Finding the Relation:.....	80
4.7 Summary	92
CHAPTER FIVE EVALUATION	93
5.1 Introduction	93
5.2 Model Evaluation	93
5.2.1 Face Validity	95
5.2.2 Similarity Measure	105
5.3 Results and Discussions	111
5.4 Summary	113
CHAPTER SIX CONCLUSION AND FUTURE WORK	114
6.1 Introduction	114
6.2 Research Summary.....	114
6.3 Limitation.....	115
6.4 Future Works	116
APPENDIX A	126
APPENDIX B	128
APPENDIX C	149
APPENDIX D	160

APPENDIX E	162
APPENDIX F	168
APPENDIX G	170



List of Tables

Table 2.1: Term Based Representation Techniques	27
Table 2.2: Graph Based Representation Techniques	40
Table 4.1 Tags Meaning	72
Table 5.1 The Participants' Selection Criteria.....	96
Table 5.2: Similarity Percentage for 20 CDM.....	110



List of Figures

Figure 2.1: Pre-processing Steps.....	14
Figure 2.2: VSM Representation of Document.	18
Figure 2.3: Sample of Dependency Graph Representation.....	37
Figure 2.4: Sample of Conceptual Graphs Representation.....	39
Figure 2.5: graph of the exemplary canonical data model.....	43
Figure 2.6: Canonical Data Model Graph Base Representation	44
Figure 3.1: The Steps of the Research Methodology	46
Figure 3.2: Research Design	48
Figure 3.2: The 20 Newsgroups.....	49
Figure 3.3: Construction of Canonical Data Model (CDM).....	55
Figure 3.4: Popularity Table	57
Figure 3.5 Jaccard similarity example	62
Figure 4.1: The Process of Constructing CDM.	66
Figure 4.2: A portion for Single Document	67
Figure 4.3: The result of splitting Process	68
Figure 4.4: The result of Tokenization Process	69
Figure 4.5: The result of removing the stop words.....	70
Figure 4.6: The result of Stemming Process	71
Figure 4.7: The result of POST Process.....	72

Figure 4.8: Flowchart for Generating the Popularity Table.....	75
Figure 4.10: Flowchart of Classes Constructing.....	78
Figure 4.11: Portion of the Classes Result.....	79
Figure 4.12: Flow Chart of Finding the Relation.....	82
Figure 4.13: Portion of the Relationship between Classes	83
Figure 4.14: Graphical Display of CDM	84
Figure 4.15: Diagram for CDM Model Creation	85
Figure 4.16: Flow Chart of constructing a CDM.....	87
Figure 4.17: The Number of Words Before and After Using Threshold.....	88
Figure 4.18: Percentage of Words Reduction	89
Figure 5.1 Face validity process	95
Figure 5.2 Percentage of response for Question 1	99
Figure 5.3 Responds percentage for Question 2	100
Figure 5.4 Responds Percentage for Question 3	101
Figure 5.5 Responds Percentage for Question 4.....	102
Figure 5.6 Responds Percentage for Question 5.....	103
Figure 5.7 Participants Responds.....	104
Figure 5.8: The Pre-Processing Step in Rapid Miner Software.....	106
Figure 5.9: The Main Process.	107
Figure 5.10: Similarity percentage of comp.sys.ibm.pc.hardware group	108
Figure 5.11: Similarity percentage of misc.forsale group.....	109

CHAPTER ONE

INTRODUCTION

1.1 Background

In the last decade, text has become the most popular tool for communication due to the rapid technological increase. Realizing that extracting useful information from text is not an easy task, there is a need to have an intelligent tool which is able to extract useful information as quick as possible and at a low cost (Jusoh & Alfawareh, 2012) and the most prominent method to handle the task is text mining (Gharehchopogh & Khalifelu, 2011). According to Fleuren and Alkema (2015), text mining is the process of extracting new knowledge from a predefined information by regulating the distance between piece of information into certain meanings.

Text mining is considered a vivid domain for research that changes the stress in text-based information to the level of exploration and analysis from the level of retrieval. It is also one of the famous way to organizing unstructured information (Patil & Saraf, 2013). Summaries of the words are derived from information to make it easy to investigate words used in the documents (Suguna & Gomathi, 2014). Organizations can explore interesting rules, models and patterns from the text in the same manner as data mining searches data in the tables (Jhanji & Garg, 2014).

The most difficult part in text mining is the complication involved in a natural language, i.e., every natural language faces some ambiguous issues in its structure of

sentences. For example, one word can be used in different meanings, and one sentence or phrase can be deciphered in numerous ways. The primary objective of text mining is the extraction of unambiguous information from ambiguous sentences (Alfawareh & Jusoh, 2013).

Apart from text mining, text representation is another critical issue in text mining. The symbolical demonstration of text documents plays a vital role in various uses of information retrieval and data mining. Thus, unorganized data needs to be converted into a structured form so that it can be easily comprehended. A number of pre-processing methods have been proposed to design an effective classification technique for the efficient representation of documents. The most dominant of them is Bag of Word (BoW) which is also known as Vector Space Model (VSM) (Harish, Guru, & Manjunath, 2010). In this method, each aspect relates to the term frequency of a matchless word in hash-table or dictionary. But this scheme has its own imperfections, e.g., lack of association with adjacent words and its meaning which includes in the document, mainly because of its excessive sparsity (Chen et al., 2013).

To surmount the above mentioned inadequacies, a method named “term weighting” is used that assigns suitable weights to the term in which the occurrence of words are represented with each entry in the document (Korde & Mahender, 2012). In addition, instead of VSM, a new method has been proposed by Jiang et al. (2010) called graph-based model that gives an accurate classification in the form of expressive

documents' encoding. The main contribution of this technique is that it has the support to classify text documents that can be identified using subgraph mining. The technique also increase computational overhead during text representation in a realistic way.

Moreover, another challenging issue arises as how text can be represented in such a way that two functions are similar, i.e., the functions can be verified by means of isomorphic representation. In the field of computer science and information technology, canonical and tractable representation of knowledge bases has shown a great concern of interest. The study of canonical representation has gained an enormous attention from the research community of Artificial Intelligence (AI) (Darwiche, 2011), where propositional representation of knowledge bases is the division of Negation Normal Form (NNF). Canonical graph-based representation makes it easy to examine that two functions are similar by verifying that they are isomorphic by representation (Dietrich & Lemcke, 2011). However, it can be a crucial element if representation is assumed to be the optimum collection of sets. Canonicity, which has primary importance in combinational equivalence analysis, is the most prominent characteristic in the canonical representation. It can be used as a tool for verifying the equivalence of arithmetic calculations in the design of dataflow (Ciesielski, Jabir & Pradhan,)

On the other hand, CDM is a prototype that is used for interconnecting various data formats (Coldicott & Lane, 2009). It is also used to achieve compatibility between

logic-based and object-oriented data models (D. Hsiao, Neuhold, & Sacks-Davis, 2014). CDM demonstrates a familiar model that can be extracted to link all local models (Hsiao, Neuhold, & Sacks-Davis, 2014). This model also provides a cohesive collection of patterns without which the combination of personalized occurrences is a difficult task. Moreover, the CDM proposes the option to obtain a particular interface that resembles to a set of statements contained in the CDM (M. Dietrich & Lemcke, 2011).

1.2 Problem Statement

Text-mining is referred to the method of extracting exciting and important information and knowledge from unstructured text (Patil & Saraf, 2013). The data that result from the pre-process step including the parsing and tokenization which reclassified and reorganized the text in to the groups that will cause the analyzed data sparse and has many null values . Sparsity is a very familiar problem in text-mining. Structural sparsity has elicited an increasing amount of attention recently; sparsity is one of the core characteristics of real-world data. For this purpose, many techniques have been developed; one of them is called Term Frequency-Inverse Document Frequency (TF-IDF), which counts weight of the words by looking to their frequency. However, if a word is found in more than one file, it is considered as an insignificant word (Hakim, Erwin, Eng, Galinium, & Muliady, 2014).

In addition, to sparsity semantic is one of the big issues in the text mining (X. Wei, Xiaofei, Lei, Quanlong, & Hao, 2001). This is due to the complexity to find the meaning of the words as the words might have different meanings. Traditional methods cannot represent the semantics of documents accurately because of the different meanings for a single word. As text documents grow exponentially, textual data is becoming more diverse in vocabulary, i.e., text documents carry information containing its meaning (semantics). Therefore there is a need to improve text representation accurately and efficiently to increase the performance of text mining techniques (Wei, Lu, Chang, Zhou, & Bao, 2015). In this research, canonical data model is proposed to overcome the problem of semantic and sparsity.

1.3 Research Questions

The researcher in this study intends to answer a number of research questions related to solving current problems of sparsity and semantic in the context of text representation. The following research questions were formed based on the problem mentioned earlier:

- i. How to adapt CDM to represent text for reducing the sparsity problem?
- ii. How to decrease semantic problem using CDM?
- iii. How to evaluate the proposed CDM?

1.4 Research Objectives

This research aims to propose a graph based on canonical data model. There are three sub-objectives for this study:

- i. To design and implement a model that represents text by adapting CDM to reduce text sparsity.
- ii. To adapt CDM for representing text to reduce semantic problem associated with documents.
- iii. To evaluate the proposed CDM.

1.5 Research Scope

This research mainly aims at providing an alternative way to categorize the words' meaning based on the schema of graph text representation. This research used CDM along with the 20 Newsgroup datasets. The datasets used in this research consists of different news stories in various areas which were categorized based on news obtained since 1987. In addition, the subjects were sorted based on experts' views of the chronological order of these stories. The 20 Newsgroup is publicly available online and was used by many previous studies to test and examine new techniques in a text document representation (Lan, Tan, Low, & Sung, 2005). These datasets were used for the processing stage in the text mining known as pre-processing step, text representation step, and finally producing a graph by using the canonical data model.

1.6 Significance of the Research

This research tries to overcome the language difficulties and text representation by adapting a Canonical Data Model (CDM) for manipulating text document to process and represent data which 80 percent of it is in unstructured form. CDM is able to represent the contents of multi documents regardless to its domain, by integrating a set of documents in one comprehensive representation. Canonical data model can be used as a general model that has potential act as a reference model for text comparison in a variety of text mining task such as (text clustering, text classification, text summarization) and can be developed to work with more complex language compositions.



CHAPTER TWO

LITERATURE REVIEW

2.1 Text Mining

Text Mining (TM) is the invention of deriving high-quality information in text documents. The main aim of TM is to make good use of information included in textual documents which exist in many forms by using several techniques such as predictive rules, invention of patterns, and connection among entities, etc. It is also known to discover appropriate and fascinating information from a large database (Bhaisare & Nayyar, 2014).

TM is described as the process of empirical analysis of data which can lead to the detection of obscure and unspecified information. In other words, it is defined as to answer those questions which are completely incomprehensible and for which the answers are now unknown (Panda, Panda, & Giridhar, 2015). The main goal of TM is to utilize textual information in the document. The textual information can be either connection among different entities and analytical rules or pattern detection in the text data (Yuanyuan & Jianhu, 2011).

From the above discussion the main theme of text mining can be reflected as:

Primarily: abstracting the characteristics of text by a method called text segmentation that is used for the transformation of text data into an organized form, and then developing the organized text with the help of any data mining system, for example, analysis of association and its categorization, and clustering, etc.

Secondly: determining new ideas concerning the composition of text, and finding the suitable association between words and sentences.

During the process of text mining analysis that consists of information extraction, information retrieval, pattern recognition, tagging many issues arise through the process because of automatic natural language processing (NLP) such as sparsity and semantic due to the ambiguity of the language.

Text mining has a number of tasks which consists of aspects related to the categorization of textual content, clustering, principle production of granular taxonomies, notion evaluation as well as modeling of entity relationship (Agnihotri, Verma & Tripathi, 2014).

2.2 Text Classification

Text classification is an important technique for organizing text documents into classes. Automatic text classification finds applicability for a number of tasks such as automated indexing of scientific articles, spam filtering, identification of document genre, classification of news articles, etc. For example, in spam filtering, we can use

text classification to assign “spam” or “non spam” label to each new message (Sebastiani, 2002)

In text classification, a text representation model is needed to represent text content. A good text representation model can help to improve classification results. As to text classification algorithms, machine learning is a common approach, which can be divided into supervised classification and unsupervised classification algorithms (Basu & Murthy, 2012).

2.2.1 Supervised Classification Algorithms

These algorithms use the training data containing labeled texts, to learn a classifier which classifies new texts. Supervised machine learning techniques like Support Vector Machine, K-Nearest Neighbors, Naive Bayes, Decision Tree, etc. are applied frequently in text classification (Aphinyanaphongs et al., 2014).

2.2.2 Unsupervised Classification Algorithms

In unsupervised learning algorithms, we have unlabeled collection of text. The aim is to cluster the texts without additional knowledge or intervention such that documents within a cluster are more similar than documents between clusters. K-Means, Hierarchical clustering, etc. are commonly used as unsupervised learning techniques in text classification (Boyer, Martínez-Trinidad, & Carrasco-Ochoa, 2014).

2.3 Text Preprocessing

Text preprocessing refers to the mechanism of processing raw data for it to be used in further processing stage. It is frequently utilized in different practices associated with data preprocessing which is initiated with transforming data into a format. Such format is commonly used to recall a certain entity easily (Abdullah & Fadhil, 2014; Reshamwala & Mahajan, 2015).

The current practices of processing text in linguistic field has revealed to be highly relying on the use of natural language in order to effectively determine the text's features in any document. However, categorizing these features in a predefined document consider as a challenge that mostly rely on the processed language along with the source of origin where the document came from. In addition, the integrity of text meaning is not always trivial. This usually exists in document that includes different human languages. Hence, it is difficult to convert raw text data when preprocessing meanings into a defined sequence at the lowest level characters. This is because of that representing single text's feature in most language's written system could results in a substantial issues related to having one or more character in a sentence exist in one or more words. As such, preprocessing of such events can help categorize the features of each character from the association with the words and sentences identified in earlier stages (Palmer, 2010). With this in mind, text preprocessing is classified into document triage and text segmentation.

2.3.1 Document Triage

Document triage refers to the main steps required in order to transform digital data into well-defined text documents. This was a slow, manual process, and these early corpora were rarely more than a few million words require a fully automated document triage process. This process can involve several steps, depending on the origin of the files being processed. First, in order for any natural language document to be machine readable, its characters must be represented in a character encoding, in which one or more bytes in a file maps to a known character. Character encoding identification determines the character encoding (or encodings) for any file and optionally converts between encodings. Second, in order to know what language-specific algorithms to apply to a document, language identification determines the natural language for a document; this step is closely linked to, but not uniquely determined by, the character encoding. Finally, text sectioning identifies the actual content within a file while discarding undesirable elements, such as images, tables, headers, links, and HTML formatting. The output of the document triage stage is a well-defined text corpus, organized by language, suitable for text segmentation and further analysis (Palmer, 2010).

2.3.2 Text Segmentation

Text segmentation however corresponds to the process of transforming text corpus into an initiated component words and sentences. It divides the sequence of

characters in a text by locating the word margins based on the range of the preprocessed word. For the purpose of segmenting computational linguistics texts, the labeled words are regularly denoted to tokens, and word segmentation is also known as tokenization. Text normalization is a related step that involves merging different written forms of a token into a canonical normalized form; for example, a document may contain the equivalent tokens “Mr.”, “Mr”, “mister”, and “Mister” that would all be normalized to a single form.

Figure 2.1 shows the preprocessing steps in which the document is firstly inserted in order to extract the features in its text as explained earlier. The next steps is segmented for the aim of extracting the longer processing units that carries one or more words in a certain sentence. Such process also concern about determining the relevant boundaries available in words of different sentences. Then, the segmented sentences are stemmed to eliminate the punctuation marks at sentence boundaries, sentence. The phrases of every sentence will be extracted when it contain less than ‘n’ terms of the original sentence. In this step, the stop word removal is applied in order to eliminate all possible stop words from the extracted sentence in order to compute the phrase important scores. This is important practice to help rank extracted phrases in both word and sentence segmentation. However, if the period for marking an abbreviation is considered a part of the abbreviation token, then, a period at the end of a sentence is usually considered a token in and of itself.

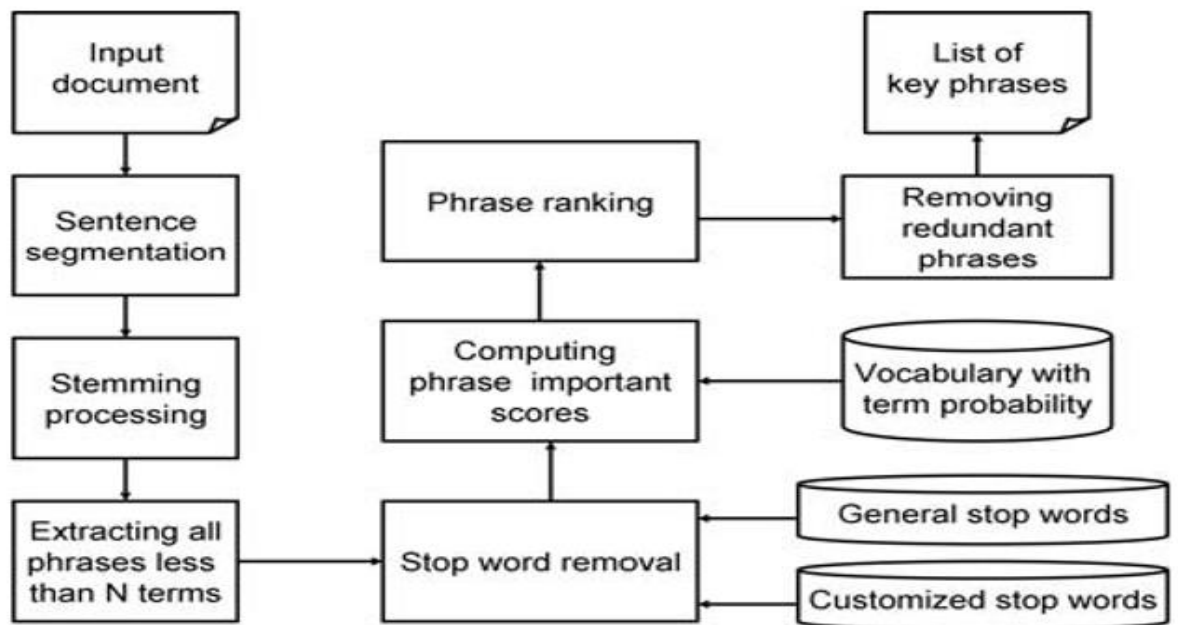


Figure 2.1: Pre-processing Steps

2.4 Text Representation Methods

To store any information in a meaningful way, text is considered as the best and common form for it. In the process of text mining, document representation is the most vital step. Therefore, the main issue is the proper demonstration of the documented knowledge that can be efficient for demonstrating the textual information in a meaningful form (Sonawane & Kulkarni, 2014). This is a very old and challenging task that has been studied for the last two decades.

In the meantime, it is one of the basic and utmost significant challenges in the field of information retrieval and processing, and yet a considerable issue (Chen & Wu, 2012). Hu and Liu (2012) have defined the representation of the text as the most

common way for shaping documents and then convert them into VSM, and deal with them through linear algebraic operations. This exemplification is named as BOW or VSM. The linguistic structure within the text is overlooked in the basic models of text representation which can lead to structural problems. In BOW model, a word is demonstrated as a separate variable which can have numeric weight of varying significance (Hu & Liu, 2012), A text representation can be efficiently divided into two categories term-based representation and graph-based representation as shown below:

2.4.1 Term Based Representation

Term-based representations can easily represent text without any prior knowledge of the text, and with suitable indices, many operations can be carried out very efficiently. The preprocessed words are called Terms and it is typically represented easily BoW by using frequencies of words as features (Cohen, 1998). There are other ways to represent text which can be classified into two categories depending on the features that been used:

1. Features may include collocations and frequent phrases.
2. Features may be constructed from statistical analysis of co-occurrences of successive words.

According to Szymański and Duch (2010), the lack of word order and simple grammatical constructions is a severe limitation of such representation. Besides that, the disadvantage of such methods is that it produces very high dimensional feature spaces and requires large training sets. Another compelling advantage of this representation is that with a good weighting scheme, the term-weight representation of documents with a document can be surprisingly effective model of its semantic content; in particular, documents with intuitively similar semantic content often have similar representations

a- Vector Space Model (VSM)

To represent a text document, Vector Space Model (VSM) is used as an algebraic model, i.e., it is used as vectors of identifiers and is commonly used for rankings in relevancy, information retrieval, pattern recognition, due to its effectiveness and easiness (Guo, Xiang & Chen, 2011; Pan, Wang & Xia, 2012; Zhou, 2010).

Any query or document can be demonstrated as vector where every dimension is characterized as a separate term. The value of a term becomes non-zero, if it is occurred in the document which is also known as term weight. Normally, terms are any keywords, a single word, or even a phrase in a long sentence. If we choose words as the terms, then the vector dimensionality is the number of words in the sentence, i.e., the number of different words happening in the corpus. To compare queries with documents, vector operations are the best tools for it. To calculate the semantic

distance, Zhou (2010) has used the VSM model in his research. According to his study, to discover an approach for Semantic Web Services and then tally these services with ontology in a hierarchical form, some amendments are needed in the matching algorithms, and hence he found through simulations that the process gets better results (Zhou, 2010).

Moreover, VSM is represented in a relational form that can be extracted from text corpus (Guo et al., 2011). It is an approach that can be signified as the weight value of each individual frequency of social objects in a text document, i.e., the relationship between text corpus and social objects can be reflected through VSM. Guo and Xiang (2011) have also concluded that deeper social relationships can be attained through VSM which is hid in the text and its corpus, and consequently, the efficiency and effectiveness of social network analysis can be highly increased included in the text corpus.

Zhang and Odbal (2012) had designed a method that aligns Mongolian-Chinese parallel text on the Web through VSM model automatically, while the representation of sentence is conceptually done by a vector of keywords obtained from the document based on VSM. They claimed through their results that the system is well enough and useful to be implemented without the interruption of any human being (G. Zhang & Odbal, 2012).

It can be concluded from the discussion that VSM is a simple and accurate model for text representation, which is based on linear algebra, and accepts calculation of an uninterrupted degree of similarity among documents and queries. But apart from its numerous advantages it also has some weaknesses, for example, finding a keyword must be matched exactly with the terms in documents. Another weak point can be word substring that can cause a false positive match (Dayananda & Shettar, 2011).

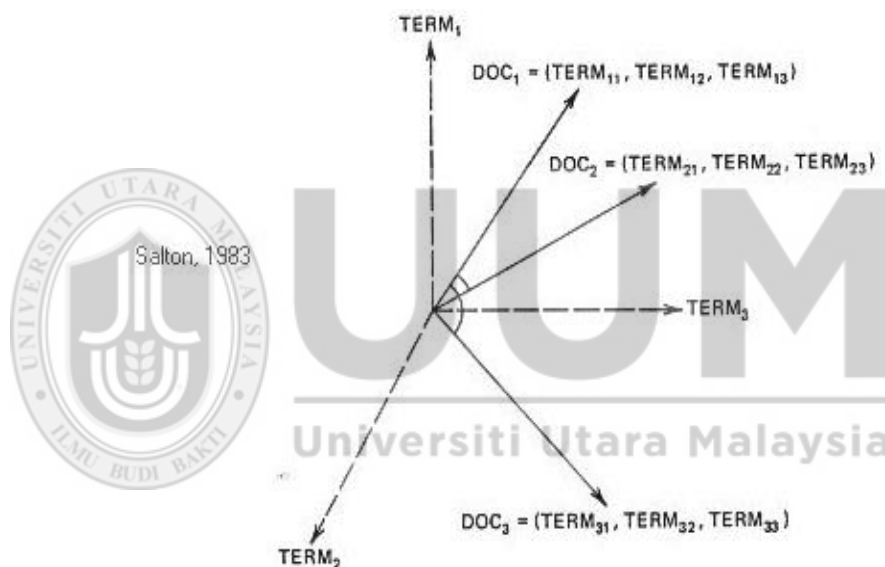


Figure 2.2: VSM Representation of Document.

b- N-GRAM Model

N-Gram Model is a kind of language model which is based on probability, that is, the next item in the text can be predicted in the form of (n-1) order. Nowadays the most commonly used models are N-Gram model and Markov chain model; both are used in communication theory, probability, computational linguistics, computational

biology, data compression, and natural language processing (Deshmukh, Deshmukh, & Deshmukh, 2014; Dey & Prukayastha, 2013; Lazim & Kamaludin, 2013; Malarvizhi & Mohana, 2014).

The idea of N-gram was first introduced by Shannon in 1951. Since then it has become the most popular method and has been used in many research fields, e.g., speech recognition and prediction, string searching, spelling checking (Shannon, 1951). For speech recognition and statistical modeling of data sequences, the N-gram model has been successfully implemented. This model consists of fast decoding, flexible, and compact algorithms that is used simultaneously for semantic/syntactic tags and word prediction (Galescu & Allen, 2001).

It is a worth noticeable that N-gram, which is a sub-sequence of n items, is used in a number of research areas. This notion can be easily understood through the following example: Given that the character sequence (for example, the sequence “for ex”), what is the probability of the next character? From training data, a probability distribution can be derived for the next character given a history of size n: $a=0.4$, $b=0.00001$, $c=0 \dots$; while the probabilities of all possible “next-letters” totals to 1.0.

Due to the open nature of language and its computational weaknesses, an unlimited number of words are available, for which independent hypotheses are made so that every word is then depended merely on the last n number of words, making it a useful Markov model (El-Shishiny & Volkov, 2011).

In addition, Francis and Nair have designed a model for Malayalam language which is based on N-gram that extracts words from the text document and have measured the relation by calculating the probability of the similar words. In actual, the designed model is meant for plagiarism detection of text as well as ideas (Francis & KN, 2014).

Two more related subjects in the area of Weighted Finite-State Transducers (WFST) based G2P conversion, have been investigated by Josef et al. (Novak, Minematsu, & Hirose, 2013). The foremost is the effect that the method employed to translate a target word to a corresponding finite-state machine (FSM) has effectiveness of downstream decoding. The next problem reflected is the impression that the system employed to demonstrate the joint n-gram model through the WFST framework that having the accuracy and swiftness of the method. It has also been shown that translating source language to target words via finite-state machines produce more accurate results.

Moreover, they have proposed two new algorithms, that make it easy to translate joint n-gram models to the corresponding WFST in such a fashion which is similar to the precise assessment through special semantics of failure-transitions (Novak et al., 2013). N-grams have a number of practical benefits for various reasons, for example, its extraction from corpus is easy, it is easy to draft prospective models for them, and they offer helpful estimation of probability for different readings of the input.

As stated by Glauco et al (Pedrosa & Traina, 2013) the principal benefit of exercising N-gram models is the easiness and the capability to improve representation of the contents successfully by simply increasing N.

Besides the advantages of N-grams, they also have some limitations, for example, if the original document is modified then it cannot perform well. Furthermore, procedures based on N-gram can be misled by changing some basic text in the document. Deletion and insertion, or even replacement of a specific token in a text can cause mismatch of at least one n-gram (Nawab, Stevenson, & Clough, 2013).

c- Term Frequency–inverse Document Frequency (TF-IDF)

TF-IDF is an arithmetical marker which is envisioned to expose, in a corpus or collection, how critical a word is to a document. In text mining and information retrieval, it is normally used as a weighting factor. If a word appears in the document again and again, the TF-IDF value is increased proportionally, however, the frequency of a word in the corpus balance it, which is useful for the adjustment of some words that appear more frequently in the text (Y. Chen, Sun, & Han, 2015; Melnikov & Vorobkalov, 2014; D. Wei, Zhang, & Zhu, 2014).

Usually, TF-IDF is used in regular text Information Retrieval (IR) in alliance with a related function to evaluate the significance of a group of documents to a query. The frequency of each lexicon in a typical TF-IDF turns out to be normal because of IDF.

IDF also decreases the frequency (value) of words that continuously appear in the group. Consequently, the importance of common words is decreased, which confirms that connection of documents become more functioning with the help of word using relatively low frequency (Schedl, 2012).

The weighting scheme of TF-IDF is used to convert a document to numeric vector from a BoW to make sure that every word in the vocabularies demonstrates a dimension. Let's take an example for better understanding, if a document needs to be selected which is quite appropriate for the query "brown cow*" among texts in English and a group of document; an easy way to start this job is to remove those documents which do not have all these three words, i.e., "in," "brown/* and "cow**.

Yet, this job leaves a number of documents. To further recognize it, each chapter must be counted (according to the frequency of its occurrence) and then add them altogether. The subsequent sum is known as the frequency of the term appears in the text.

However, the term "the,"* which is very familiar, may give result in a stress on unsuitable documents that appear more often in the utilization of the word "to" without supplying enough weight to the strong terms "brown** and "cow".

The words "brown" and "cow", which are less common, and the word "the" which is a common one, cannot be distinguished because of non-related and related terms in

the documents. Therefore, the frequency factor for the opposite document is contained to decrease the terms which appear most often in the group of documents and enhance the weight of the terms that rarely appear. The major weak point in TF-IDF is that it ignores important meaning of the words which are linked among words and/or word semantics. This situation creates several problems, the most vivid of which is sparsity.

To develop an operative scheme for clustering, the word which can occur frequently, must be normal regardless of the frequency of occurrence in the document. Vector space-based TF-IDF technique is the most popular representation used for text processing. Because TF and IDF are related to the significance of the word, therefore, they are considered as of extreme importance (X. Huang & Wu, 2013).

If a word occurs more than one time in a document, as expected, then it is reflected as the utmost vital word and is assigned a high score. Similarly, if a word occurs in a number of documents, it cannot be considered as a distinctive identifier, therefore, it is assigned a low score. On the other hand, most frequently used terms, for example, "for" and "the", which occur in several documents, are scaled down. While those terms that occur most often in a particular document, are scaled up.

In the field of text mining, TF-IDF is considered as the major scheme of usage. The main advantage to utilize this scheme is that it permits for the approximation of the importance of a word in a text document. The number of repeated words in a single

document is considered as the frequency of that word. IDF includes a calculation of the typical importance of a word. In a mathematical representation, the weight of a term in TF-IDF is denoted as:

$$W_i = TF_i * \log \left(\frac{D}{DF_i} \right) \quad (2.1)$$

In equation, TF_i means the frequency of appearance of term i in a particular document; D is the number of documents in the text; and DF_i is the frequency of document that includes term i . Thus, $\log (D/DF_i)$ means frequency of the opposite document.

The increase in the importance of a term occurs if the weighting approach is used for the word repetition, but, this increase is rewarded by the duplication of the word in a particular corpus. The weighting method is called the precision of clustering, which is used commonly for the improvement of text (Liu & Yang, 2012).

TF-IDF weighting may be obtained from a matrix know as regularity information. When the weights of the vectors are decided, they are applied on the frequency of text. TF-IDF is a word-weighting utility used to represent a document (Aggarwal & Zhai, 2012). One of the major weaknesses of TF-IDF is that it ignores significant meanings which are associated with words and/or their semantic and matches documents based only on the frequency of words.

(Ramos, 2003) have mentioned in their study that the significance of TF-IDF stipulates the term in the corpus of documentation which may be possibly more appropriate if it is used in a question. TF-IDF may be used to compute the significance of each term in the document regarding the percentage of the opposite frequency of the term in a compulsory document in proportion to the texts in which they occur. This reflects the strong association of terms in the required document they appear in, representing that if a specific lexicon occurs in the question that can be used by the speaker. The proof is provided by this condition that question retrieval can be highly improved by simple algorithms which efficiently sort the related lexicons.

Yun-tao, Ling and Yong-cheng (2005) have mentioned that the novel and underlined TF-IDF scheme enhances the precision of text classification, which uses supportive and confidential words. Further they have mentioned that synonyms decided by a lexicon are handled in the enhanced TF-IDF scheme that produces inspiring results.

A neural network-based learning component is used by Pandit et al. (2008) along with TF-IDF, which restricts the study to the practice of inductive techniques and tries to augment the TF-IDF vector with extra words. These words can be obtained by exploiting a neural network-based component expert in the pairs of words and documents to show that the word in the pair lies in the document's context.

Numerous research work has been done on the utilization reduction schemes for dimensionality, for example, probabilistic hidden semantic analysis (Hofmann, 2001), which pursues a k-generative model for the appearance of words in a particular text or document. This system tries to substitute the VSM with a latent-space model (LSM).

Huang et al (Huang & Wu, 2013) have proposed an enhanced TF-IDF scheme for the resolution of the low-slung accuracy of micro-blog commercial word mining and application in term of computation. To apply this scheme, it involves categorization of a huge number of micro blog information lies in a particular model and then allots word weights for the modules using a framework called Hadoop distributed. According to their simulated results, the use of this enhanced TF-IDF scheme in micro-blog commercial word mining is operative and of great concern.

In the recent study of Juan et al. (Juan Ramos, Sangwan & Tim Zwietasch,), they have shown some advantages of TF-IDF, i.e., they consider TF-IDF as one of the proficient and easy schemes for word matching in a query to documents which are related to that query. They have collected some data and have applied TF-IDF on it, and have found it highly relevant that returns documents to a particular query. TF-IDF can find documents that include some specific knowledge related to the query, if a query is sent for a particular topic. Moreover, TF-IDF is very easy to be encoded; making it suitable for establishing the foundation for more complex schemes and query retrieval methods (Ramos, 2003; Sangwan, 2014; Zwietasch, 2014).

Besides, it also has some weaknesses in case if there is any synonym word, TF-IDF cannot identify its relationship with other terms. Also, it cannot make difference between the singular and plural words and hence, somewhat decreases the value of words. It may also create an unavoidable problem for a large collection of documents (Ramos, 2003; Sangwan, 2014; Zwietasch, 2014).

However, sparsity is the major problem using TF-IDF, which is a prominent issue in the field of statistical modeling. In the last few years, structural sparsity has gained an enormous attention from the research community, because it is one of the main advantages of the real-world data.

Table 2.1: Term Based Representation Techniques

Term based representation	Author and year	Advantages	Drawback
Vector Space Model (VSM)	(Dayananda & Shettar, 2011)	<ul style="list-style-type: none"> • Simple and accurate for text representation. • Accepts calculation of an uninterrupted degree of similarity among documents and queries 	<ul style="list-style-type: none"> • Finding a keyword must be matched exactly with the terms in documents • Word substring that can cause a false positive match
N-GRAM Model	(Nawab et al., 2013)	<ul style="list-style-type: none"> • Faster computation time 	<ul style="list-style-type: none"> • The model cannot perform well if the original document is modified (such as: deletion, insertion or replacement). • Needs large memory allocation.
TF-IDF	(Ramos, 2003)	<ul style="list-style-type: none"> • Can be applied to both frequencies of data and presence of data 	<ul style="list-style-type: none"> • Ignores important semantic links between words and/or word meanings.

2.4.2 Graph Based Representation

The graphical representation of text has been reported by many researchers in the last decade. The main purpose of this interest is that the existing studies lack semantics in the text representation. Representing the text in graphical format consists of the analysis concepts, graphs focusing on ideas, and conceptual representation of graphs. For graph partitioning, some schemes have been proposed that are also suitable for action graphs.

For this reason, cutting edges of graphs identify the clusters in such form that the addition of the total number of weights of the edges, which need to be cut, is reduced. In graph, each individual node represents a document. The edge occurs between two nodes if the resemblance between two documents lies in distinct clusters. Similarly, those edges which are included in one cluster will have more weight than the edges lie in different clusters. Every graph-based method can create the base in a different way and in turn, these methods can also use methods of partitioning in a different manner (Kaur & Kaur, 2013).

Text representation has used by many researchers for the classification method. An electronic mail which is denoted as a graph rather than a simplified BoW has improved possibility of gaining high precision as compared to other schemes.

Chakravarthy et al. (2010) designed a scheme based on graph theory uses two graph representation schemes, known as domain-independent schemes, for covering the

text, i.e., email and websites. For providing domain's focus, on the basis of domain information, the graph representation schemes are chosen. For graph representation, basically three techniques are used which are explained briefly in the following sub-sections.

a- Ontology

Ontology can be roughly understood as an active, controlled vocabulary for defining a number of concepts or objects included in a domain or process, and their interrelationships. It can be further believed as a documented knowledge-representation method that consists of policies for encoding the style in which knowledge is characterized, together with the instructions which allow automated reasoning with respect to the concepts or objects represented (Gardner, 2007). The term ontology can be a single named concept defining an entity or object. An idea can consist of a group of words and related relevance weights, word localization statistics, and co-occurrence which define a topic.

Furthermore, an ontology is the actual method of a perception of the knowledge of a community of an area, the knowledge may be taken and provide to both human beings and machines by an ontology. An ontology can have a number of methods, however inevitably it will contain a lexis of terms, and may be some descriptions of their significance (Stevens, Goble & Bechhofer, 2000).

In addition, Gruber explains ontology as “the requirement of conceptualizations used to assist humans and programs share knowledge”. The conceptualization is the expression of awareness of the world in terms of entities (for example, objects, the connections they have and the restrictions among those objects). The specification is the symbolical form of this conceptualization in a real shape. The major stage in this order is to set the conceptualization in an informative image of a specific language. The purpose is to develop such kind of vocabulary and structure of semantic for the exchange of knowledge regarding an area that is considered as an agreed-upon method (Miguelanez et al., 2008).

The major elements of ontology are axioms, relations, ideas, relations, and instances. An idea shows that a class or group of objects or entities lies within an area. Concept is divided into two types, i.e., primitive concept and defined concept. The first one is that which only has essential situations for affiliation with a group (in terms of its properties), while the second one is that which is indispensable to be defined for an object to be a member of the group.

Moreover, relation defines the communications between ideas or a property of idea. It can also be divided into two general groups, i.e., Taxonomies, and Associative relationships. Taxonomy is the structured ideas into sub-ideas or super-ideas tree form. The utmost popular methods are specialization relationships and partitive relationships. Specialization relationships are well-known relationships, while partitive relationships are used to define ideas which are belonging to other ideas.

The second group which is called “Associative relationships” is the one which connect ideas across tree forms. Popular examples found related to associative relationships are the following:

Nominative relationships, that define the name of ideas, and Locative relationships which are used to define the position of an idea. Associative relationship demonstrates the processes and functions of an idea and other advantages of that specific idea.

Even though, some shared experience are available for the usage and development of ontologies, no field of ontological engineering is available that can be compared with the knowledge engineering. Specifically, at present, no standardized practices are available with the help of which ontology can be built. This type of practice will consist of a group of stages that lie during the construction of ontologies, strategies and rules to help in the particular phases, and the life-cycle of an ontology that denotes the associations among phases. The most famous ontology development strategies were designed by Gruber, which inspire the progress of more recyclable ontologies. In the last years, a tremendous effort has been seen for the development of a wide-range methodology for ontology (Capasso, 2008).

Likewise, the performance of underpinning algorithms for ontology based on DBSCAN and k-means clustering is compared by Punitha (Sureka & Punitha, 2012). It is used to give the idea of weight, which is computing by deciding the association

coefficient of the word and the possibility of the idea. Different tests have been done for the evaluation of performance. Their outcomes present that the presence of ontology amplify effectiveness of clustering; the DBSCAN algorithm which is based on ontology, exhibit the best functioning among the other ontology-based procedures, for example, k-means algorithms (Sureka & Punitha, 2012).

Another text mining ontology-based approach has been developed by Ma et al. (2012), that gather original research motivations based on the resemblance in the research domain. According to their results, the approach can be operational and effective in the compilation of Chinese and English texts. They have conducted a research on ontology that is used as the categorization and development of specific ideas in the arrangements for several areas of research (Ma et al., 2012).

The basic limitation related to this type of ontology is the failure to connect different areas of subjects within a domain. The second limitation can be incompetency of linking ontology terms for deducing associations or objects in a large data base.

Although current ontology systems commonly propose specifications of a specific domain in the form of entity names or nouns, mostly they have the deficiency of well-organized group of semantic relationships in the form of verbs and association between entity objects. While a number of research studies are focusing on this type of text representation, yet another limitation of existing ontology methods is

generally the lack of names, i.e., the names which are related to all ideas and information included in the group of documents.

For example, Bloedorn et al. (2005) developed a framework for text mining based on ontology, named OTTO that turns around the KANO model, to decide the communication among different ontologies. The research community is trying to design a system that can help to make better use of the existing ontology systems to cope with their concept hierarchies and lexica for the improvement of results both in organized and unorganized sets, for example, the utilization of Reuters-21578 corpus using WordNet as ontology (Bloehdorn et al., 2005).

Gardner et al. (2007) derived an information management system (MIS) and designed a system to combine organized and unorganized texts through an ontological approach, which further contains procedures for the validation, creation, enhancement, and combination of ontologies for life science informatics and other specialties (Gardner et al., 2007).

b- Dependency Graph

A dependency graph is a kind of directed graph which represents the needs of various objects. Attaining an estimation instruction or the non-attendance of an inference order which admires the reliance from the graph is conceivable without a

doubt (Balmas, 2004). This kind of graph is a suitable illustration of the craving association.

The mentioned graph consists of a set of suggestions, i.e., node, the values of nodes, and the connecting arc in the form of a continuous set of dependency associations that limit the task of assurances. Zimmermann et al (2008) designed a dependency graph as a directed graph $G=(V,E)$, while V is a group of nodes and $E \subseteq V \times V$ is a group of edges also known as dependency. Dietrich et al (2008) defined the searching of object dependency called ODEM, which stand for Object Dependency Exploration Model. The purpose of this method is to mine the reliance of graph. The dependency graph consists of nodes used for the representation of classes. The mentioned nodes have comments which control their grouping, (i.e., interface, annotation, and class, etc.), abstraction, visualization, and certainty.

Every individual node also comprises a record of one-sided associations, i.e., dependencies, along with the full name of a given class denoted by an association of dependency annotation (e.g., usages, ranges, or tools). This method increases the vision, and therefore creates the form of a graph less obscured. Om et al (Qu, Qiu, Sun, & Wang, 2008) suggested a new scheme known as feature event dependency graph (FEDG), which is able to give information more proficiently than CGs. Another scheme is recommended by Dietrich et al (2008) for the usage of the Girvan-Newman clustering algorithm to compute the integrated form of packages. This technique is thought as a novel scheme to examine the dependency graphs of

object- oriented programs. The proposed method is thought to be significant for the professionals of software engineering as they redesign the element restrictions in software to enhance the stage of protection. The usage of the mentioned method claimed to be applicable, but the vision of the scheme in ways where a huge amount of nodes happens needs extra expansion.

Patel et al (2009) proposed a novel clustering method which integrates static dependencies and dynamic analysis. The dependency graph consists of the link representation for the association among different classes, and the nodes are known as the classes. The graph is a directed graph with at most two edges between two classes, and all edges have the same weight. The extraction of structural relationship is an automatic task supported by numerous tools. Extraction tools vary in their support of numerous expertise.

Mitchell and Mancoridis (2010) used a method of source code investigation for the construction of a dependency graph which shows the components of the system and component-level inter- relationships. The graph was then utilized as input to the group of instrument that divides the graph. The outcomes were showed in a bunching graph through graph imagining.

The experimental results obtained by Wang et al. (2011) presented that the dependency graph system displays the best functioning in a required text grouping among systems which focus on the BoW prototype. This system may also recognize

fundamental associations and enhances the functionality of the comparison quantity between documents.

Beck and Diehl (2013) proposed a scheme that includes the integration of dependency graphs afore presenting assembling a practical group of processes, for example, union, weighted union, and junction on the group of edges. The scholars determined that joining both schemes enhances the total amount of grouping.

This segment shows the kinds of techniques utilized in the document illustration and their effect on the grouping procedure. Investigators who utilized these techniques and their findings that recognized the features and drawbacks of each scheme were also shown. In the existing studies, the scheme of the reliance graph for the illustration of documents was utilized to show if this scheme reveals greater precision in the illustration of documents compared with other techniques. Table 2.2 shows the strengths and weaknesses of each individual kind of technique used for the illustration of these documents.

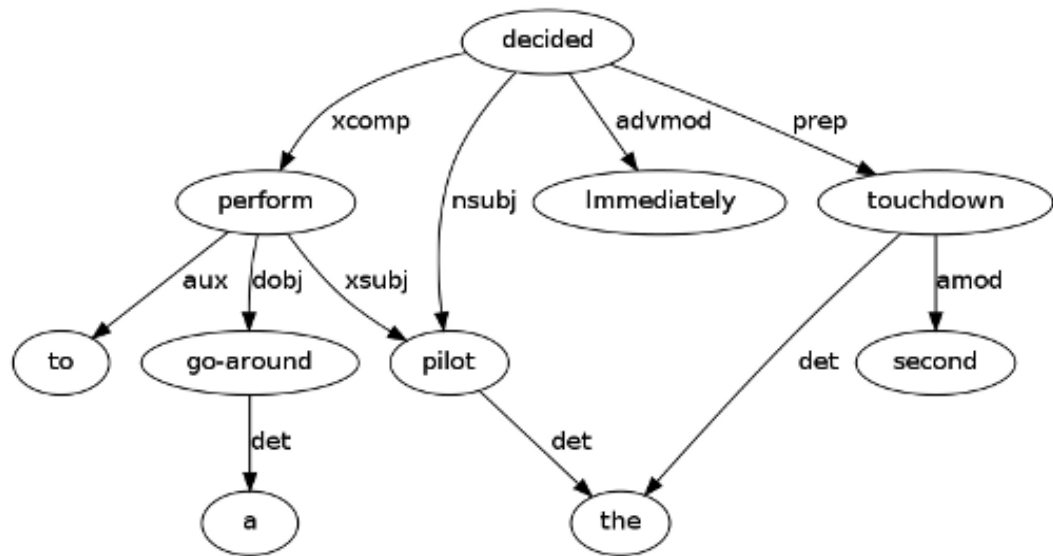


Figure 2.3: Sample of Dependency Graph Representation.

c- Conceptual Graphs (CGs)

Conceptual graphs (CGs) are used for knowledge representation which consists of a language. According to Sowa and Way (1986), the basic use of these graphs is in the field of philosophy and linguistics psychology. CG is also used to represent structure of knowledge at semantic level, which includes two basic components, i.e., relation and concept. Therefore, CG is divided into two equal parts, and is connected together in a finite manner. For the graph representation, a group of nodes and edges for vertices are used. CG is also used to represent well-defined and exact information. For matching graph, the most frequently used technique is CG because its produced results are unailing for numerous reasons. The method in which the data is compared in a particular text is simplified with the help of CG for text representation.

(Boroch & Heger, 2010) in their research on “Vehicle Environment Description and Interpretation using Conceptual Graphs” claimed that video-based driver assistance systems are an important part of the vehicle safety strategy. The conceptual graphs application was used in this work to describe the relations between the ego vehicle and other objects, as well as the relations of those objects between each other. Based on this information the behavior of the objects could be predicted. Conceptual Graphs has many advantages as they have reasoning mechanisms which allow versatile querying algorithms, they are expressive enough to be able to represent the rules associated with extracted data, easy to plug in on top of existing ontologies due to the distinction between ontological knowledge (the support) and factual knowledge (bipartite graph).

Beside the benefits it also has a number of drawbacks (Delugach & Lampkin, 2000; Andrews & Polovina, 2011) such as lack of direct support for automated knowledge acquisition except for relying on an experienced CG analyst to transcribe graphs based on interviews and documents, there is no well-established technique for building conceptual graphs to represent an expert's knowledge. Another limitation of conceptual graphs it does not support for probabilistic reasoning that is to be expected since they rely on a deontic logic of Boolean valued propositions.

Conceptual Graphs have been used to represent Financial Text by exploits the constituent structure of sentences and general English grammar rules to perform the transformation, Creating a computerized generator for the aim of translating some

sort of sentence structure into a conceptual graph that consists of categories brings several obstructions when extracting textual content in addition to facilitates the particular implementation. Additionally, such process causes extra challenges from the perspectives of building conceptual graphs as a way for presenting textual contents in addition to formalize homogeneity (Kamaruddin, Bakar, Hamdan & Nor, 2008).

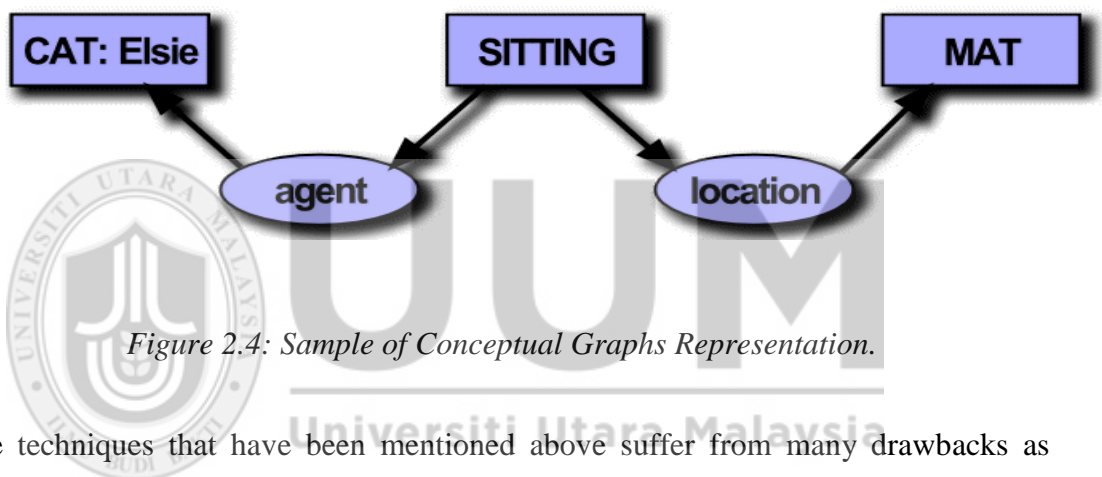


Figure 2.4: Sample of Conceptual Graphs Representation.

The techniques that have been mentioned above suffer from many drawbacks as shown in Table 2.1 and Table 2.2. Complexity, sparsity, and semantic are the most important issues among all techniques drawbacks. This research tries to propose a canonical data model to overcome these issues.

Table 2.2: Graph Based Representation Techniques

graph based representation	Author and year	Advantages	Drawback
Ontology	Gardner, (2007)	<ul style="list-style-type: none"> • Helps to visualize data 	<ul style="list-style-type: none"> • Needs distance measure to compare ontology. • The inability to link different areas of specialty within a discipline. • Ignored some words during the construction of the ontology.
Conceptual Graphs	Wang & Liu, (2008)	<ul style="list-style-type: none"> • Able to capture relation between words 	<ul style="list-style-type: none"> • Comparing graphs are computationally complex
Dependency Graph	Wang et al., (2011)	<ul style="list-style-type: none"> • Discover causal relationships and improvement the performance of similarity measure between texts 	<ul style="list-style-type: none"> • The visualization needs to be improved

2.5 Canonical Data Model

Canonical Data Model (CDM) is a design prototype used to transfer between different forms of data (Coldicott & Lane, 2009). It is a model that combine the ideas of logic-based and object-oriented data models for the purpose of attaining compatibility (Hsiao, Neuhold, & Sacks-Davis, 2014). According to Roebuck (2012), “Any model which is by nature canonical is known as canonical data model”, that is a model available in the simplest form based on any standard and provides integration solution for applications. The CDM is based upon the following core principle (Comer, 2010).

1. Understandability. The CDM reflects the common vocabulary of the domain. It does not introduce obtuse abstractions. It is modularly organized into readily understood views.
2. Independence. The CDM is independent of any specific application, but rather synthesizes the data integration requirements of all relevant applications.
3. Immutable Kernel. The core structures (entities and relationships) of the CDM are logically common across all applications and are not expected to change.
4. Extensibility. The CDM is designed to be extended for particular installations and for future change and evolution.
5. Commutative Mapping. The process of mapping specific information models into the CDM must preserve data and operations.
6. Separation of Concerns. The CDM logically separates the data by who controls or owns the data.

The CDM is used to offer an amalgamated group of representations without the integration derivation of modified occurrences is not feasible. Additionally, current techniques are not traditionally accepted and hence, it is difficult to obtain mappings from current methods to a new modified instance. The CDM provides the opportunity to obtain a particular boundary that agrees upon a group of techniques available in the CDM (Dietrich & Lemcke, 2011).

According to Gonzalez, Martí and Kruchten (2011), the CDM is used to envision the interoperation of simulation in a combined system for disaster response simulation. The CDM includes entities that demonstrate a conceptual topological design of the physical layer of the OSI model and illustrate the communication between simulators generally, which provides substituting information between the simulators at any given time. It is therefore concluded that the CDM is useful in envisioning the complication for the design of the disaster response simulator (Gonzalez et al., 2011).

In addition, Dietrich and Lemcke (2011) developed a CDM with the perception of attaining the input schemas from CIDX and Noris. They claim that their designed CDM ideally manages the research for procedures that iteratively generate a model from the input schemas. This can then be associated with the ideal CDM for further estimation. The main goal of the CDM is to take the advantage of all schemas in a way called semantic equivalence. Simultaneously, from the equivalent parts, the CDM should record the deviation of all schemas. The CDM can therefore be recognized as the resolution of all schemas shown in Figure 2.5. (Dietrich & Lemcke, 2011).

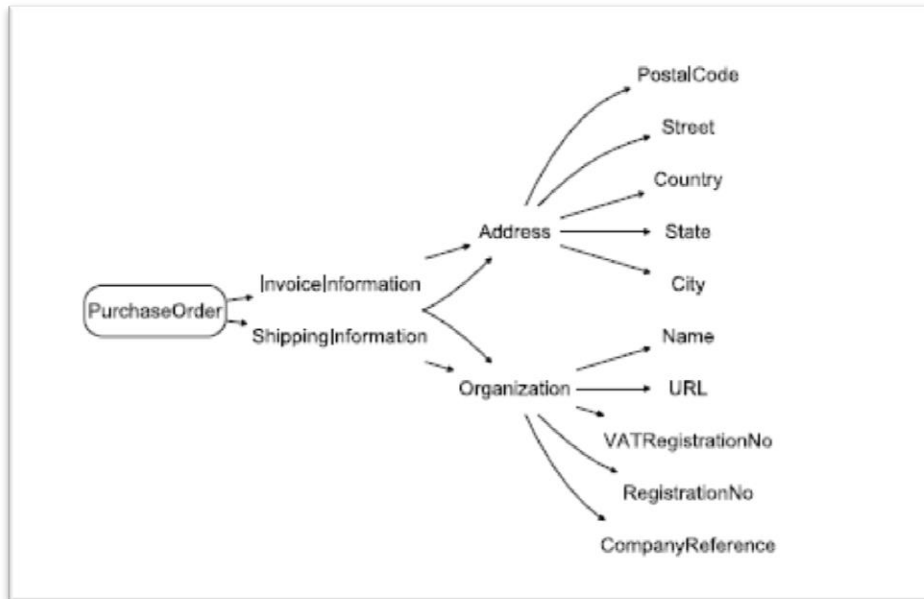


Figure 2.5: graph of the exemplary canonical data model

Bloechle et al. (2006) used CDM in their work for representing the structured content of PDF document. They developed a special system which called XCDF with specific well-defined properties to represent structured electronic document by using canonical data model to represent the original content in respect to structures and annotations. They concluded that CDM is the suitable solution to represent the data by giving the labels to each component of the document after the textual content of the document is been preprocessed and segmented to the blocks.

According to Dietrich et al. (2010), CDM has been used in their project called Warp 10 which help organizations access to protected activities. The authors considered the use of Warp10 to help in importing import different schemas linked to a certain

business model. They also proposed mapping these models with a CDM based on the Core Components Technical Specification (CCTS) standard. Therefore, CDM have been used for mapping between source language and the target language of the data from different business sources with different languages. Based on the result, CDM have increases the implementation effort and pre-processing step from the source language to the target language.

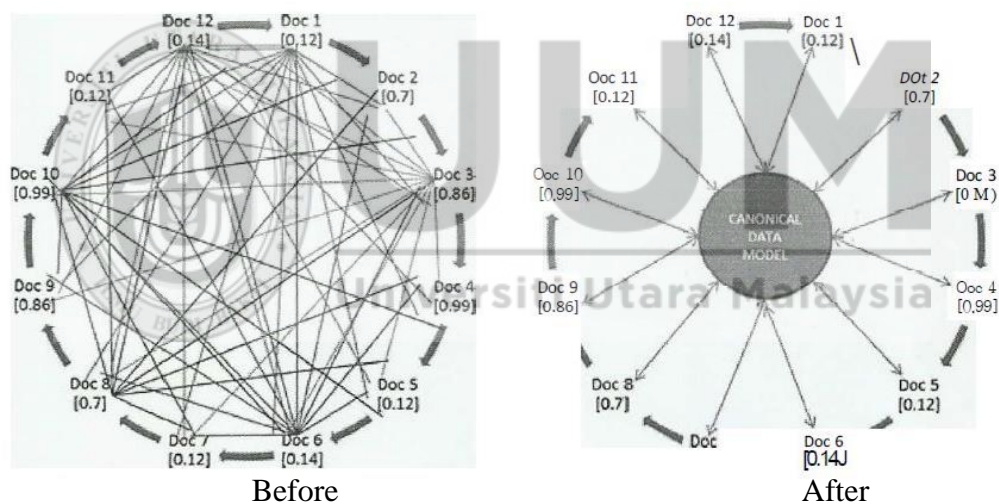


Figure 2.6: Canonical Data Model Graph Base Representation

Figure 2.6 illustrates the relation between objects before and after using the CDM and it also shows that how it reduces the density among objects.

2.6 Summary

This chapter provides a detailed study of the graph types and a review of previous studies on these methods as well as the strengths and weaknesses of each type. This chapter also presents one of the most common text representation methods known as TF-IDF, which has been gaining a tremendous attraction by the research community. Moreover, a number of text representation schemes are also discussed, with focus on TF-IDF and the Canonical Data Model (CDM) for the resolution of the sparsity problem, which is the focus of this study.



CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

A research is a systematic method to solve real world problems based on organized planning. It is also the application of logic and objectivity in order to achieve the aim of the research. This chapter explains the research methodology that was used to accomplish the objectives of this research as depicted in Figure 3.1 below.

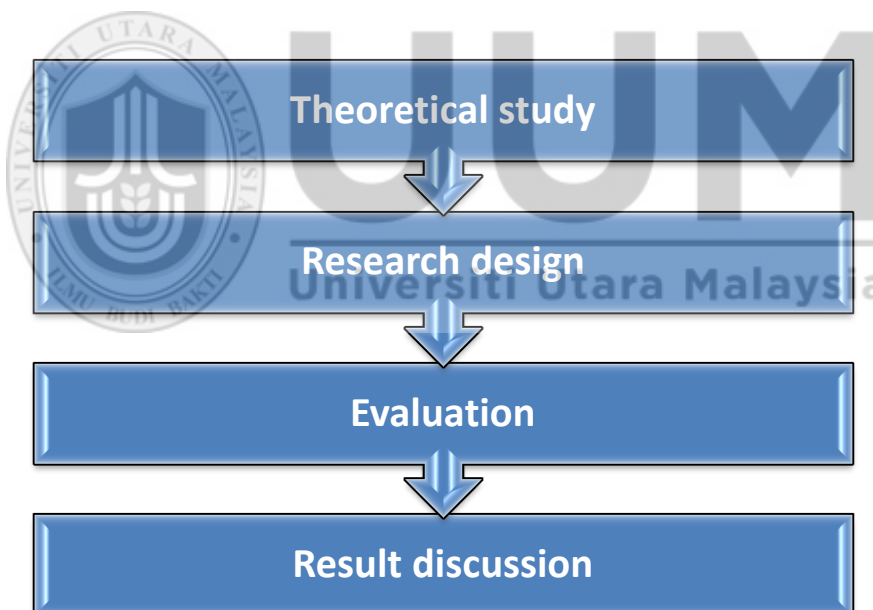


Figure 3.1: The Steps of the Research Methodology

3.2 Theoretical Study

The analysis of previous researches shows that some of the methods used for text representation have some disadvantages that need to be taken into consideration for the efficiency of text representation. Many researchers have been trying for the last decade, as discussed in Chapter Two, to improve text representation techniques either by improving the current strategies or proposing some new mining schemes. Majority of the problems faced by text representation schemes include, high dimensions of representation, overlooking at the link and semantic association that happens between the terms in a document.

3.3 Research Design

The proposed scheme, which is called canonical data model, combines semantic weighting and concept weighting using graph-based text representation. The scheme consists of the following stages:

- i. Document Collection
- ii. Document Pre-processing
- iii. Constructing Canonical Data Model
- iv. Canonical Data Model

Document Pre-processing consists of four steps, i.e., splitting, tokenization, remove stop words, stemming and post-tagging. While text representation schemes include parsing, semantic and construction of canonical data model. The concept of these steps is illustrated in Figure 3.2.

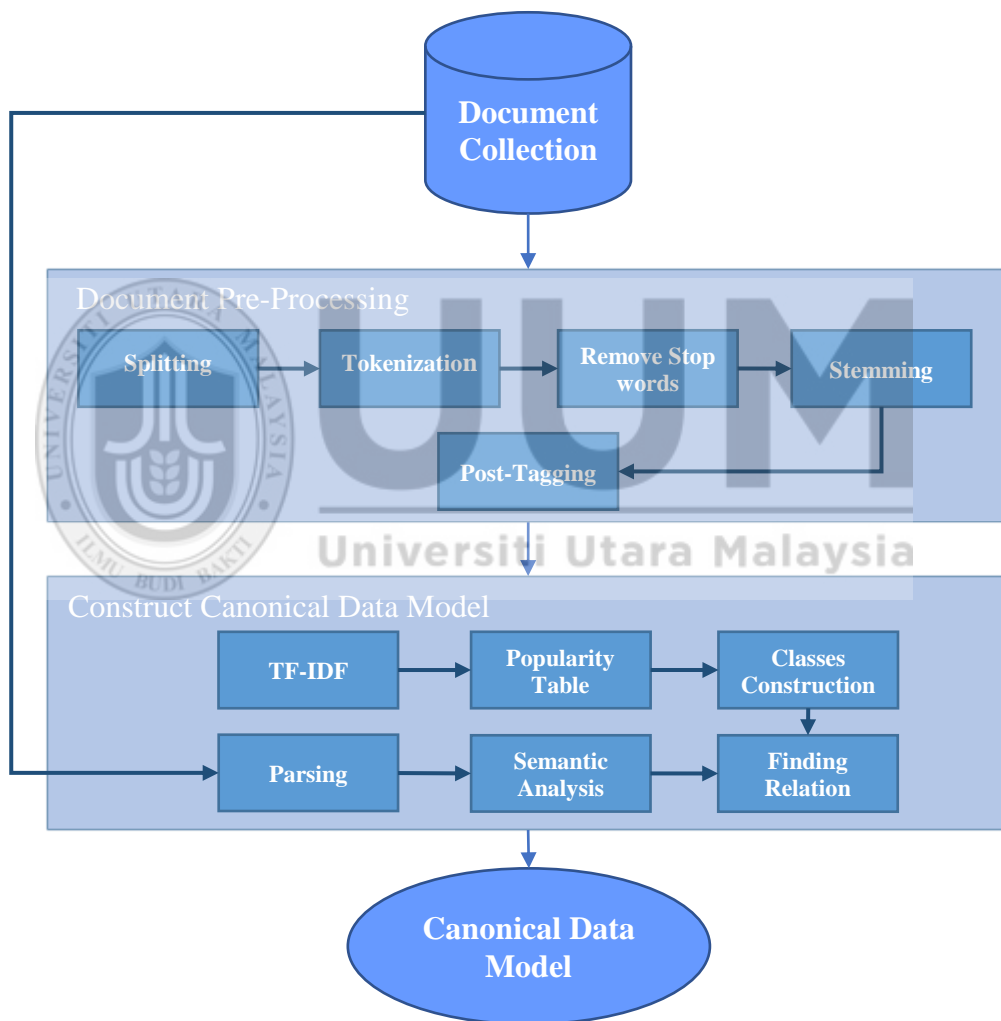



Figure 3.2: Research Design

3.3.1 Document Collection

This research used the 20 newsgroups datasets, which were compiled by Lang in 1995 as a set of 20,000 newsgroup documents. The datasets were logically divided into 20 newsgroups, where each group consist of 1,000 document which belong to a unique topic. A few of these newsgroups are tightly related to each other, which can be refereed in (comp.sys.ibm.pc.hardware/comp.sys.mac.hardware), while other newsgroups are found to be irrelevant for this study, for example (misc.forsale/soc.religion.christian). The list of 20 newsgroups is demonstrated in Figure 3.2.



Subgroup Computer	Subgroup Sport	Subgroup Science	Subgroup Politics	Subgroup Religions	Subgroup Others
comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian	misc.forsale

Figure 3.2: The 20 Newsgroups

3.3.2 Document Pre-processing

In Pre-processing step, several methods were performing to transmute text data into a structure template of documents for text mining purposes. The main functions of these processes were to achieve the fundamental quality or the main terms from news online text documents and to improve the similarity between document and words as

well as the similarity between category and words. Most documents include some unnecessary words, which negatively affected the representation of documents.

In this research, Python language was used for the execution of the proposed scheme. For the cleaning-up of the text of phrases or words, there were 4 important steps to be followed, namely: Splitting the documents, Tokenization, Remove stop words, Stemming, and POS tagging

a- Splitting of documents

The first phase in pre-processing is the splitting of every document into sentences by separating the textual content (usually string) into different sentences. The common practices used in the case of textual content written in English, punctuation is used to support the process of identifying the full stop character. Such aspect must be considered as it is a standard role in English writing which may or may not also terminate a sentence. For instance, when textual contains "Mr.", it does not mean that ‘.’ Ends the sentence, similar with “Smith went to the shops in Jones Street.” As with word segmentation, not all written languages contain punctuation characters which are useful for approximating sentence boundaries (Durugkar, 2013).

b- Tokenization

Tokenization is another method used to divide the string of text into symbols, words, phrases, or other significant components known as tokens. To further process the list

of tokens, tokenization provides the input data to be parsed for text mining. Tokenization is very helpful in linguistics, as it provides a foundation for text segmentation, and in computer science it provides part of lexical analysis (Fares, Oepen & Zhang, 2013).

Tokenization is used to decide the borders of sentence and to divide the text into a string of individual tokens (words) by eliminating unnecessary punctuation marks. The text is divided into words with line breaks, spaces, and other kinds of word breakdown in English or any other language.

Text document should be appropriately recorded for processing. A maximum number of punctuation marks are isolated from their connecting words, and productions are divided into integral morphemes such that each morpheme is tagged disjointedly. For example, "we're" can be tokenized to "we" and "are". This tokenization separately allows additional analysis of each element distinctly. Therefore, "we" can be in the subject noun phrase, while "are" is the head of the leading verb phrase. Tokenization of subtleties for ellipsis, hyphens, dashes, and dots likewise exist.

Parsing or other types of operations can be performed on isolated lines in the text by the Tokenizer. Tokenizer classify different lines and their association among them. For instance, the existence of an English letter article "the" in the last part of a line appends proof for the presence of text or a sentential zone of lines (Huang, Šimon, Hsieh & Prévot, 2007).

c- Remove stop words

Stop words are a set of words which are filtered out before or after processing natural language data, usually refer to the most common words in a language such as (a , the , an ..) .Stop words are basically a set of commonly used words in any language, not just English. The reason why stop words are critical to many applications is that, if we remove the words that are very commonly used in a given language, we can focus on the important words instead. A sample of stop words for English language can be found in **Appendix A**.

d- Stemming

Stemming is the reduction process of modified words to their root, stem, or base form or generally, a written word structure. The process of stemming is used to decide domain vocabulary in domain analysis (Dolamic & Savoy, 2009). For instance, stemming systems must have to determine the string "cats" ("probably "catlike", "catty" etc."), which is based on the root "cat", "stemmer", "stemming", "stemmed" as depends on "stem".

The words can be reduced with the help of stemming, for example "playing", "played", and "player" to the root word, "play". On the other hand, " explore", "explored", "explores", "exploring", "explore" reduce to the stem "explore"

(describing the case in which the stem itself is not a word or root) but "exploration" /explorations" reduce to the stem "exploration".

This research selected the Porter Stemmer scheme because it is easier to implement and reported to be very efficient by several previous researchers (Christopher, Prabhakar & Hinrich, 2008; Hull, 1996; Karaa & Ben, 2013). There is no origin for this linguistic method that is made by supposing that does not have a stem dictionary. However, a clear list of suffixes is provided to the program (Sureka & Punitha, 2012).

e- Post-Tagging

POS tagging or (word-category) is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition, as well as its context (Chekima, On, Alfred, Soon, & Anthony, 2012). POS tagging is done in the context of computational linguistics, using algorithms which associate discrete terms, as well as hidden parts of speech, in accordance with a set of descriptive tags (Kumar, Khulbe & Dhami, 2012).

The main job of Part-of-speech (POS) tagger is to assign the potential syntax classes of words that can be discovered in the document for the main parts of speech in English language. It has been observed by (Segond, Schiller, Grefenstette & Chanod, 1997) that POS tagging can solve ambiguity found in the meaning of words up to a

specific level, for example, 40% in their simulated analysis. The procedure of recognizing these words and their semantics can be prolonged to identify familiar terms or expressions. The identification of Phrases may happen at a number of stages to know the significance of lexicons and their connections. This process can be made as a portion of data in the knowledge base to permit the addition of words' structures. The main parts of speech in English are (verb, adverb, adjective, noun, pronoun, determiner, conjunct, interjection, and preposition). The process are tagged all the words into it corresponding part of speech but in this research only deal with (noun and verb) only.

3.3.3 Constructing Canonical Data Model

Previously documents were used to be denoted as "bag of words" (BOW model) regardless of their meaning (semantic). It is one of the major challenges that how to represent text document and also very crucial for achieving the desired results. This research focuses on the canonical data model (CDM) to resolve this issue, where every group of documents is represented as a CDM. This research is trying to represent the documents into a graph that include the relation between each word and sentence with each other. In CDM, the nodes are well-matched with words while the edge represents the association among different pairs of classes. This method manages how to convert a text document into graphs. In order to develop graph-base representation model by using the CDM that can achieve the main research objective, the CDM were designed and implemented by several steps as shown in Figure (3.3)

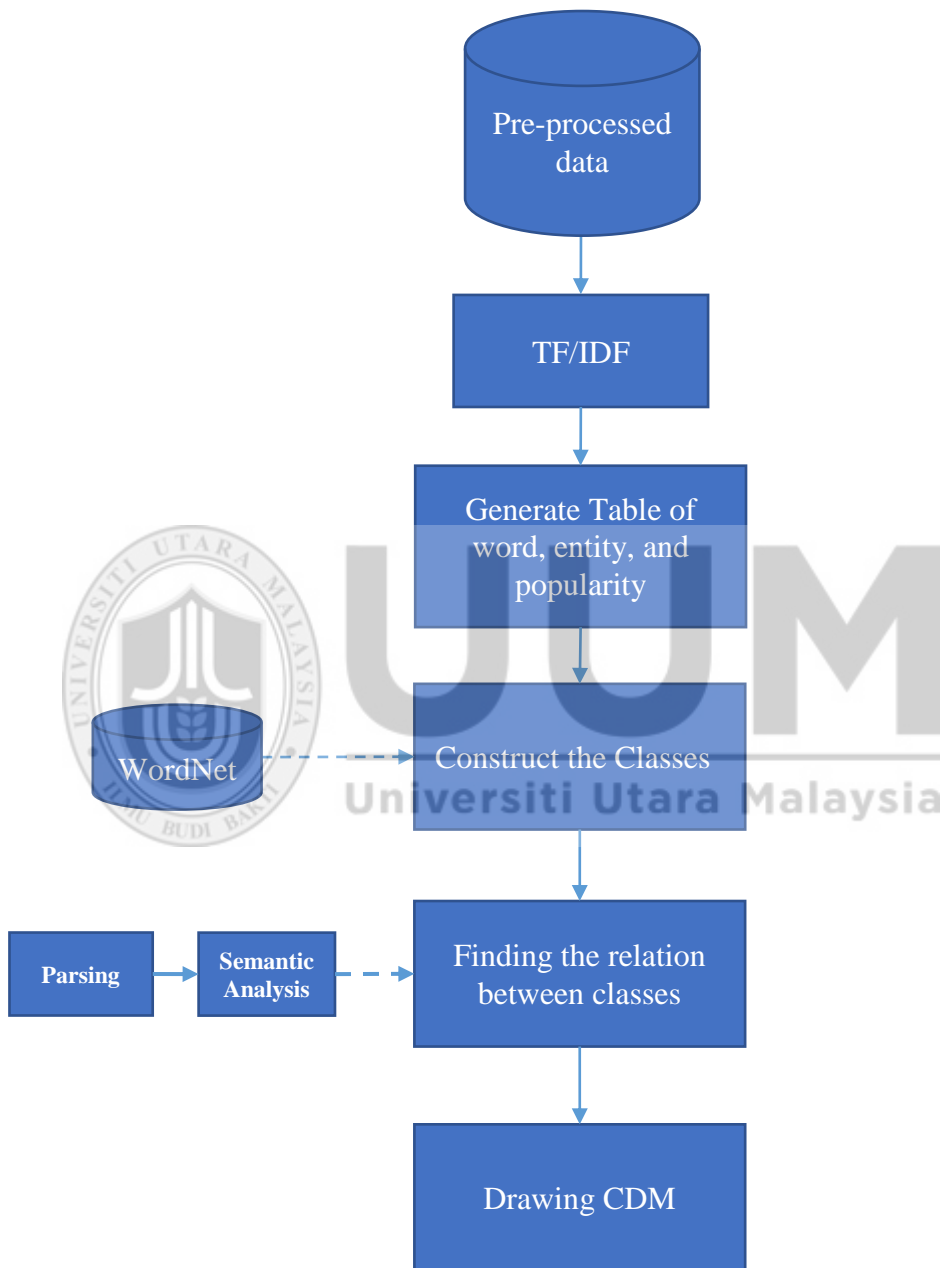


Figure 3.3: Construction of Canonical Data Model (CDM)

There were four main in designing steps to construct CDM in this research, namely: Popularity table, Classes construct, Finding the relations and Drawing CDM, which will be described below in detail:

1- Popularity table:

The data that have been produced from TF/IDF were constructed as a weighting table and were used to generate the popularity table, which contains the information of the category of each word such as(noun, verb). As mentioned before TF/IDF generate a table that contains each word and its weight based on the specific calculation and formula that has been establish as TF-IDF threshold. Please be noted that all values fell below this threshold are skipped.

$$threshold = (MaxTFIDF * tDoc) \quad (3.1)$$

Where *MaxTFIDF* is the maximum value for TF-IDF found in the dataset. And *tDoc* is a user defined parameter which indicates the percentage value with respect to *MaxTF-IDF* in order to filtering low weight words, the value of *tDoc* is (0.3) and this value was chosen after many tries (0.6,0.5,0.4) for number of reasons .As described before, the popularity table will be generated to contain each word, weight and type, as shown in the Figure 3.4 below.

Popularity Table		
Word	Weight	Type
Cat	20	N
Mate	15	N
Site	10	V

Figure 3.4: Popularity Table

2- Construct the Classes

Based on the outcomes from the previous step which was the Popularity Table, the words were grouped into classes and each class was assigned a unique identifier. This step deals with words which categorized as (noun) from the popularity table and searched for similar meaning for the noun to build the class. This step was performed by using the WordNet (Specia & Motta, 2006) to check the words meaning .

3- Relationships between classes

In this step, the relationships between classes were formed depending on the words that were categorized as verbs in the Popularity Table. In order to set these relationships parsing and semantic analysis were used and these process were explained in subsections below, the result from parsing is shaped as sentence, this result were used as input to the semantic analysis process, the process to find the relation was done by checking words that tagged as nouns before and after the verbs. (Specia & Motta, 2006).

a- Parsing

This process is used to parse sentences to recognize their syntactic arrangement. To achieve this goal, the "Link Grammar Parser" (LGP) is applied. The syntactical association between words in a particular sentence is provided by this parser because this depends on dependency grammar which is further based on the most dominant and commonly used grammar in programming languages called context-free grammars. Due to these reasons, LGP is simpler to run than the more complex parsing methods. Also, this can concurrently provide a much powerful structure of semantic than the standard context-free parsers (Suchanek, Ifrim, & Weikum, 2006).

A parser is required to achieve the association between words with the innovative sentences. The main objective of a parser is to investigate the input sentence and to generate the resultant (desired) parse tree as the output.

For example: ['the', ' priest ', ' read ', ' the ', ' bible ']

The result after parsing: (S(NP the (N priest)) (VP (v read) (NP the (N bible))))

b- Semantic Analysis

The processor of a language should perform various special functions mainly based on semantic analysis as well as syntax analysis. The main goal of semantic analysis is two-fold, i.e., to inspect whether a series of words (for example, a sentence) is well-shaped and to form it into a group that demonstrates the syntactic association between different number of words. Semantic analysis make up the most complex phase of language processing as they build up on results of the parse tree . Based on the knowledge about the structure of words and sentences, the meaning of words, phrases, sentences and texts is stipulated. To perform semantic analysis predicate logic is used, in which properties of sets of objects can be expressed via predicates, logical connectives, and quantifiers. This is done by providing a “syntax” (i.e. how elements are combined to form logical expressions) and “semantics” (the interpretation of what these expressions mean within the logical system). The example of predicate logic representations are represent the semantic interpretation or meaning of the sentences.

Sentence: the priest read the bible

The Semantics result: $x.(\text{priest}(x) \ \& \ \text{read}(x, \text{bible}))$

[For x being priest ,x read bible]

4- Drawing CDM:

This is the final step in constructing the CDM, where the graph were generated by drawing all the classes involved and the relationships between them.

3.4 Evaluation

Evaluation is a systematic and crucial part of a research. It is important step to ensure that research objectives are achieved and it provides more information for further enhancement. To guarantee that the proposed research model captures what it is intended to do, it needs to be evaluated. The effect of assessment process is important in determining the model consistency and validity. In order to perform this function, evaluation of this research model was divided into two stages. The first stage was evaluation of the model by using Face Validation. While, the second stage was to measure the similarity between the CDM results and the original text by applying the Jaccard similarity.

Stage A: Face validation

Face validity involved asking individuals that has knowledge about the systems or the subjects that are being researched about. More formally, face validity is defined as the degree to which test respondents view the content of a test and its items as relevant to the context in which the test is being administered (Sargent, 2005). In this research, face validity were conducted with five English language (experts) lecturers

in UUM language centre. (Shneiderman, 1992) suggests that having between three to five experts participating adequate. Accordingly, this research manages to engage with the five experts to evaluate CDM (Siti Mahfuzah, 2011). To determine whether they are correct and reasonable accepted. Interview was used to accomplish this part of evaluation.

Stage B: **Similarity Measurement by Jaccard similarity**

It is a statistic methods used for comparing the similarity of sample sets. The Jaccard measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (3.2)$$

This compression was ranged from 0 to 1 and the result was 1 when the documents were typically similar and it was 0 when the documents were totally dissimilar and values between 0 and 1 representing a degree of similarity.(J. Zhang & Rasmussen, 2001).

Equation 3.2 represents the equation to measure similarity and in this case :

A: present the test set while B: present the CDM. The calculation of words is to identify similarity. To clearly illustrate the technique used, Figure 3.5 shows an example to explain the equation.

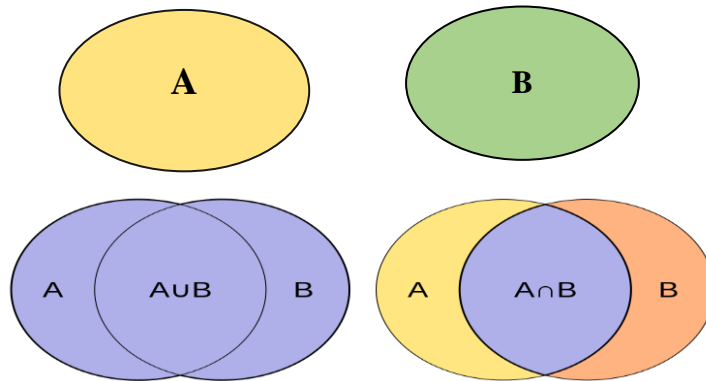


Figure 3.5 Jaccard similarity example

3.5 Summary

This chapter provided a methodology that were used in this research and it also presented the steps that have been used to design and implement the canonical data model (CDM), which has been gaining a tremendous attraction by the research community. In addition, the evaluations methods use to validate the CDM also are discussed in this chapter.

CHAPTER FOUR

CONSTRUCTING CANONICAL DATA MODEL (CDM)

4.1 Introduction

This chapter provides the implementation details of the proposed canonical data model (CDM) in this study. Figure 4.1 illustrates the processes that consists of several steps involved during the CDM construction and will be explained in the following sections.

4.2 Data Set

There are a number of standard data sources available online that are being used when testing natural language algorithms. One of the most studied data set is the Twenty Newsgroups (20NG) data set, which can be found at (<http://qwone.com/~jason/20Newsgroups>). The data can be organized in many different ways based on the specific requirements of the study. In addition to the 20NG data, a corpus of English was required. This was obtained through the use of the set of corpora that are available as part of the python **nlk** package. The details of the processes are described in the Implementation subsection and some portion of sample data from each one of the groups used in this study can be found in **Appendix B**.

4.3 Implementation Tool

This research chose Python as the implementation language, primarily because of its ease of use. Another reason was due the availability of a powerful natural language processing toolkit written in pure python. Python Version 3.4 was used as it is the most popular version of the 3 series 3 released. The following Python packages were used: **setuptools** for installation support, **tkinter** for graphics built upon the Tcl/Tk tools, **numpy** for numerical analysis and **nlTK** for natural language support with a graphical interface. In the final version of the tool, critical class/relationship data was written to an external file named **class.dat**. This was done so that external tools can be used for further processing. The format of this file was extremely simple. It contains a series of class names, with relationship names given with enclosing { } brackets.

This research deployed a single external tool called **Graphviz** as a drawing package, which can be found at www.graphviz.org. (Note that the external **Graphviz** toolkit was used, not the python **graphviz** package). This choice was made to allow for flexibility, so that additional external tools could be added to the project without the need to change the python code. A helper program, **makedot**, was used to convert between the class.dat format and the dot format used by **Graphviz**. Once the resulting **class.dot** was obtained, it can be converted into many formats. In this research it was used to generate PDFs corresponding to the CDM for a specific newsgroup.

In addition to this code, a natural language corpus must be loaded. This was done at the python command line by importing **tkinter**, **numpy**, and **nlk**, and then issuing the command **nlk.download()**. This opens a graphical window showing available corpora. The “**all corpora**” item should be chosen and downloaded. This only needs to be done once, not on every invocation of the python code.

4.4 Main Implementation Steps

The construction of CDM consisted of several phases including syntactic and semantic analysis, which involved the following two phases:

a- Pre – processing phase

- Input the documents.
- Split the documents into lines.
- Tokenize the documents so that each token represent word.
- Remove the stop words.
- Stem every word to its root.
- Tag the words.

b- Constructing the Canonical Data Model phase

- Create TF_IDF
- Create the popularity table
- Classes construction
- Finding the relation

- Parsing
- Semantic analysis

The result from the above phases (a- Pre – processing) and (b- Constructing the Canonical Data Model) had used to draw the CDM .

Figure 4.1 illustrates the process of constructing CDM and detail explanations for every phase are found in the following sections.

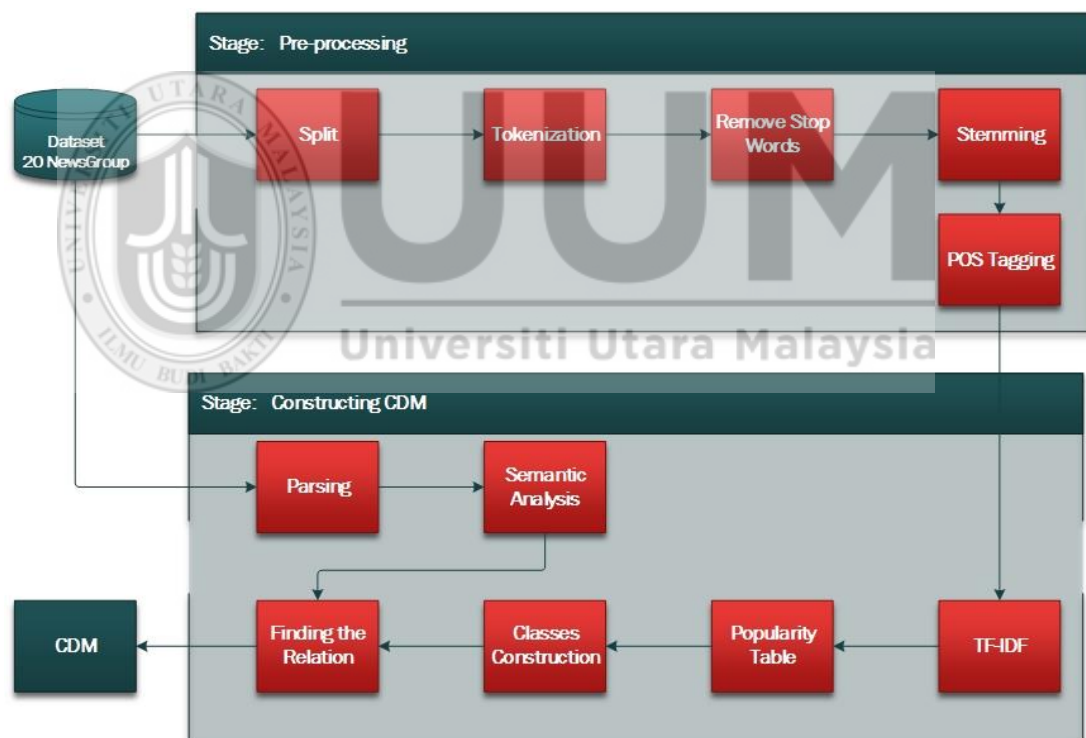


Figure 4.1: The Process of Constructing CDM.

4.5 Document Pre-Processing

Document pre-processing was the first stage and it consist of a number of: (a) splitting (b) tokenization (c) remove stop words (d) stemming (e) part of speech tagging (post). The sections below explain each one of these process in detail. Figure 4.2 shows an example of one of the document from the 20 newsgroup data called *talk.religion.misc*. The (*talk.religion.misc*) group was chosen as an example to show the result in all the processes to construct CDM.

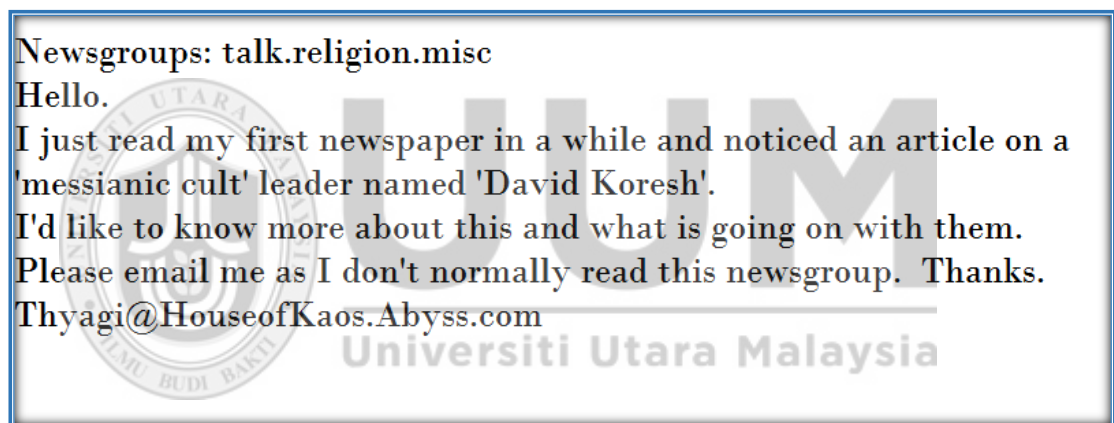


Figure 4.2: A portion for Single Document

4.5.1 Splitting

Splitting was the first process in the pre-processing stage, where the document was split into several sentences by using nltk package in python as describe below. Figure 4.3 shows the result for splitting process.

The Splitting Process

```
Import nltk /* import nltk package*/  
Open and read(document.txt)  
split=nltk.split(document.txt)  
Print (split )
```

```
['\nNewsgroups: talk.religion.misc\nHello.', "I just read my first newspaper in a  
while and noticed an article on a\n'messianic cult' leader named 'David  
Koresh'.", "I'd like to know more about this and what is going on with them.",  
"Please email me as I don't normally read this newsgroup.", 'Thanks.',  
'Thyagi@HouseofKaos.Abyss.com\n']
```

Figure 4.3: The result of splitting Process

4.5.2 Tokenization

This step tokenized the document so that each token represents a word. The tokenization was part of the **nltk** package, and thus had rules for handling various forms of quoted words. For example, tokenization handles the fact that “isn't” is a contraction of “is not”. The tokenization process is explained below and Figure 4.4 shows the result.

The Tokenization Process

```
Import nltk /* import nltk package*/  
Open and read(document.txt)  
token=nltk.word_tokenize(document.txt)
```


Print (token)

```
['Newsgroups', ':', 'talk.religion.misc', 'Hello', '.', 'I', 'just', 'read', 'my', 'first',  
'newspaper', 'in', 'a', 'while', 'and', 'noticed', 'an', 'article', 'on', 'a', '"messianic",  
'cult', '"', 'leader', 'named', '"David"', 'Koresh', '"', '.', 'I', "'d", 'like', 'to', 'know',  
'more', 'about', 'this', 'and', 'what', 'is', 'going', 'on', 'with', 'them', '.', 'Please',  
'email', 'me', 'as', 'I', 'do', "n't", 'normally', 'read', 'this', 'newsgroup', '.',  
'Thanks', '.', 'Thyagi', '@', 'HouseofKaos.Abyss.com']
```

Figure 4.4: The result of Tokenization Process

4.5.3 Remove Stop Words

This step removed stop words set, which was basically a common words such as (“the”, “a”, “an”, “etc.”). The main reason to remove them was to focus on the more important words instead of them. Once again, stop word removal was handled by a utility within the **nlTK** package. This process is shown in the steps below and Figure 4.5 shows the result of removing the stop word.

The stop word Process

```
Import nltk /* import nltk packeg*/  
from nltk.corpus import stopwords /*import stopwords package*/  
Open and read(document.txt)  
Set (stopwords("english"))  
token=nltk.word_tokenize(document.txt)  
for (each word in token)  
    if (word not in stopwords)  
        put (word) in filtered_words[ ]
```

Print (filtered_words)

```
['Newsgroups', ':', 'talk.religion.misc', 'Hello', ':', 'I', 'read', 'first', 'newspaper',  
'noticed', 'article', "'messianic'", 'cult', '""', 'leader', 'named', "'David'", 'Koresh',  
'""', ':', 'I', "'d'", 'like', 'know', 'going', ':', 'Please', 'email', 'I', "n't", 'normally',  
'read', 'newsgroup', ':', 'Thanks', ':', 'Thyagi', '@', 'HouseofKaos.Abyss.com']
```

Figure 4.5: The result of removing the stop words

4.5.4 Stemming

This process derived every word to its root, where many words were derived from the same root. The derivations were generated through appended affixes. This was very important technique for NLP that helped to reduce the size of indexing terms. This was done by reducing a variant of words form to it common root by removing the affixes from the words and reducing them to their word base. In this research, Porter Stemmer was used due to the excellence trade between speed, readability and accuracy. The process is explained below and Figure 4.6 shows the result of stemming process.

The Stemming Process

```
Import nltk /* import nltk packeg*/  
import os /*import os packge*/  
from nltk.corpus import stem /*import stem packge*/  
Open and read(document.txt)  
stemmer =stem.porter stemmer()
```

```
for (each word in document.txt)
    Print (stemmer.stem(word))
```

```
Newsgroup: talk.religion.misc Hello .I read first newspap notic articl 'messian
cult ' leader name 'David Koresh ' . I 'd like know go .Pleas email I n't normal
read newsgroup .Thank. Thyagi @HouseofKaos.Abyss.com
```

Figure 4.6: The result of Stemming Process

4.5.5 Part of speech tagging (post)

This process was one of the most important processes for text analysis tasks. It was used to classify word into their part-of-speech and mark-up the word according to tag set which is a collection of tags used for the POS tagging. The process of post is explained in below and Figure 4.7 shows the result of POST process.

The Post Process

```
Import nltk /* import nltk packeg*/
Open and read(document.txt)
Token =nltk.word_tokenize(document)
Post_word=nltk.pos_tag(token)
Print (Post_word )
```

```
[('Newsgroups', 'NNS'), (':', ':'), ('talk.religion.misc', 'JJ'), ('Hello', 'NNP'), (',', ','), ('I', 'PRP'), ('read', 'VBP'), ('first', 'JJ'), ('newspaper', 'NN'), ('noticed', 'VBN'), ('article', 'NN'), ('"messianic"', 'JJ'), ('cult', 'NN'), ('"', '"'), ('leader', 'NN'), ('named', 'VBD'), ('"David"', 'JJ'), ('Koresh', 'NNP'), ('"', '"'), (',', ','), ('I', 'PRP'), ('"d"', 'MD'), ('like', 'VB'), ('know', 'RB'), ('going', 'VBG'), (',', ','), ('Please', 'NNP'), ('email', 'NN'), ('I', 'PRP'), ('"n"', 'RB'), ('normally', 'RB'), ('read', 'VB'), ('newsgroup', 'NN'), (',', ','), ('Thanks', 'NNS'), (',', ','), ('Thyagi', 'NNP'), ('@', 'NNP'), ('HouseofKaos.Abyss.com', 'NNP')]
```

Figure 4.7: The result of POST Process

The table below explains the meaning of each tag in Figure 4.7. The words that tagged as (NN) and (VB) were the only type of tag that had used in this research as shown in Figure 4.9 and all the other type of tagged had ignored.

Table 4.1 Tags Meaning

Tag	Meaning	Tag	Meaning	Tag	Meaning
CC	Coordinating conjunction	CD	Cardinal number	DT	Determiner
EX	Existential there	FW	Foreign word	IN	Preposition or subordinating conjunction
JJ	Adjective	JJR	Adjective, comparative	JJS	Adjective, superlative
LS	List item marker	MD	Modal	NN	Noun, singular or mass
NNS	Noun, plural	NNP	Proper noun, singular	NNPS	Proper noun, plural
PDT	Predeterminer	POS	Possessive ending	PRP	Personal pronoun
PRP\$	Possessive pronoun	RB	Adverb	RBR	Adverb, comparative
RBS	Adverb, superlative	RP	Particle	SYM	Symbol
TO	to	UH	Interjection	VB	Verb, base form
VBD	Verb, past tense	VBG	Verb, gerund or present participle	VBN	Verb, past participle
VBP	Verb, non-3rd person singular present	VBZ	Verb, 3rd person singular present	WDT	Wh-determiner
WP	Wh-pronoun	WP\$	Possessive wh-pronoun	WRB	Wh-adverb

4.6 Constructing a CDM

Construction of CDM was done through a number of stages, each one of these stages is explained in detail. And *talk.religion.misc* group was used as an example to show the result of all CDM construction phases.

4.6.1 Generate popularity Table by using TF-IDF:

TF-IDF was used to find the weight for each word, where the weight was used to create a popularity table. A threshold equation was applied to reduce the number of words, where only the words that had scores above the threshold was used to generate the popularity table. While, the words whose weight were less than the threshold were ignored. After the process, the research found the reduction of the words were approximately from 17505 to 15480 for all the groups. The *TF-IDF* is one of the core methods in this research and it described in more detail.

- 1- (*TF*) The term frequency is the number of times a constituent of the term is found in one or more documents. And it compute by applying $TF(t) = (\text{Number of times term } t \text{ appears in a document}) / (\text{Total number of terms in the document})$.
- 2- (*IDF*) The inverse document frequency is calculated how important a term is, and calculated based on the relative occurrence of terms in the documents $IDF(t) = \log e (\text{Total number of documents} / \text{Number of documents with term } t \text{ in it})$.

And by multiplying the both *TF* and *IDF* we can get the $TF-IDF = TF * IDF$.

- 3- *TF-IDF* is applied to get the weight for each word in the group and the highest

weight was (10.3) for most of the groups.

- 4- (0.3) value is Selected for the (***Tdoc***) part from the threshold equation which is the constant part , this value is used for many reasons (a) for instant by applied (0.6) value the number of words were very limited.(b) it effect the number of classes element (c) influence the relation between classes . After many try's (0.6, 0.5,0.4) (0.3) value was decided to applied as the constant value for the equation: (***threshold = max TF-IDF * Tdoc***). Figure 4.8 shows the above process in flowchart.



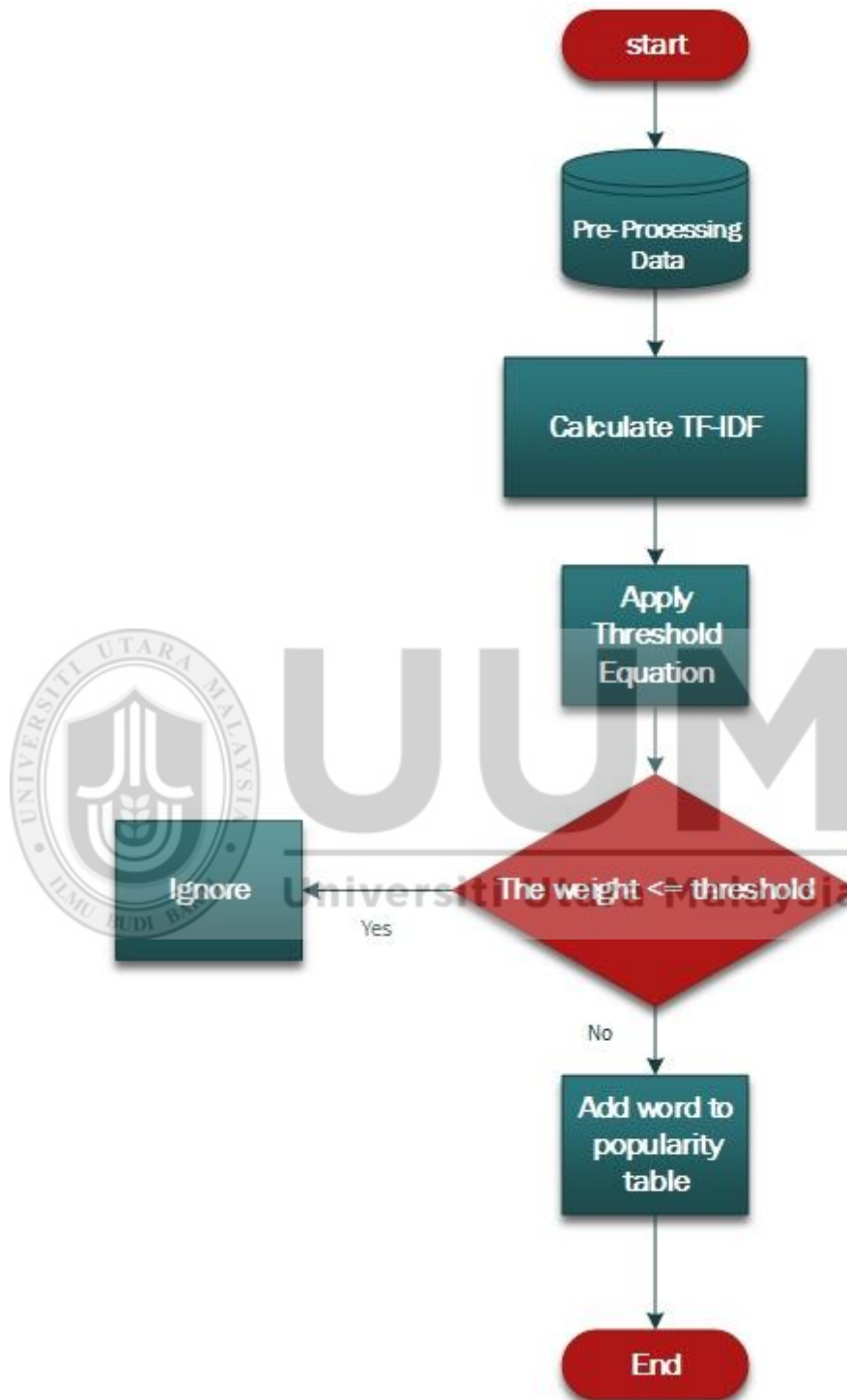


Figure 4.8: Flowchart for Generating the Popularity Table

Once the TF-IDF was calculated for each term, a weight table was created. Three categories (word, weight, type) appeared when the set of documents has been fully processed and the weight table was sorted numerically. This research used only the words that tagged as (*verb and noun*) to deal with and it's shown in Figure 4.9. A cut-off threshold was applied which discards all those terms whose weight fell below the threshold. A portion of the weight table is shown below in Figure 4.9.

```
20. Talk.religion.misc group
====***** Popularity Table *****====
compass: 9.46275390178012: NN
cols: 7.139861892300007: NN
vertex: 6.488778434090527: NN
distort: 6.432399260440207: VB
peg: 6.426301496111489: NN
archive: 6.352372352181458: VB
analysis: 6.174782811845632: NN
tracer: 6.102334228452575: NN
```

Figure 4.9: Portion of the popularity table

4.6.2 Classes Constructing:

In this step the words were grouped into classes and each class was assigned a unique identifier, In order to perform this step, the result of popularity table was needed to build the classes, only the words that were tagged as nouns from the previous process (post of tagging) were select and used to carry out this part of CDM construction. The steps below explain the Classes Construction process.

1. Choose the words that tagged as a noun from the popularity table.
2. Check if the word has meaning.
3. If the word exists in WordNet set the word as class_ID. Else the word ignored the word.
4. Go to the next word.
5. If the word already set as class_ID append the word to the class attribute (if the word carry the same meaning with the class_ID), else set the word as class_ID.

The entire process step explained in in the flowchart in Figure 4.10.

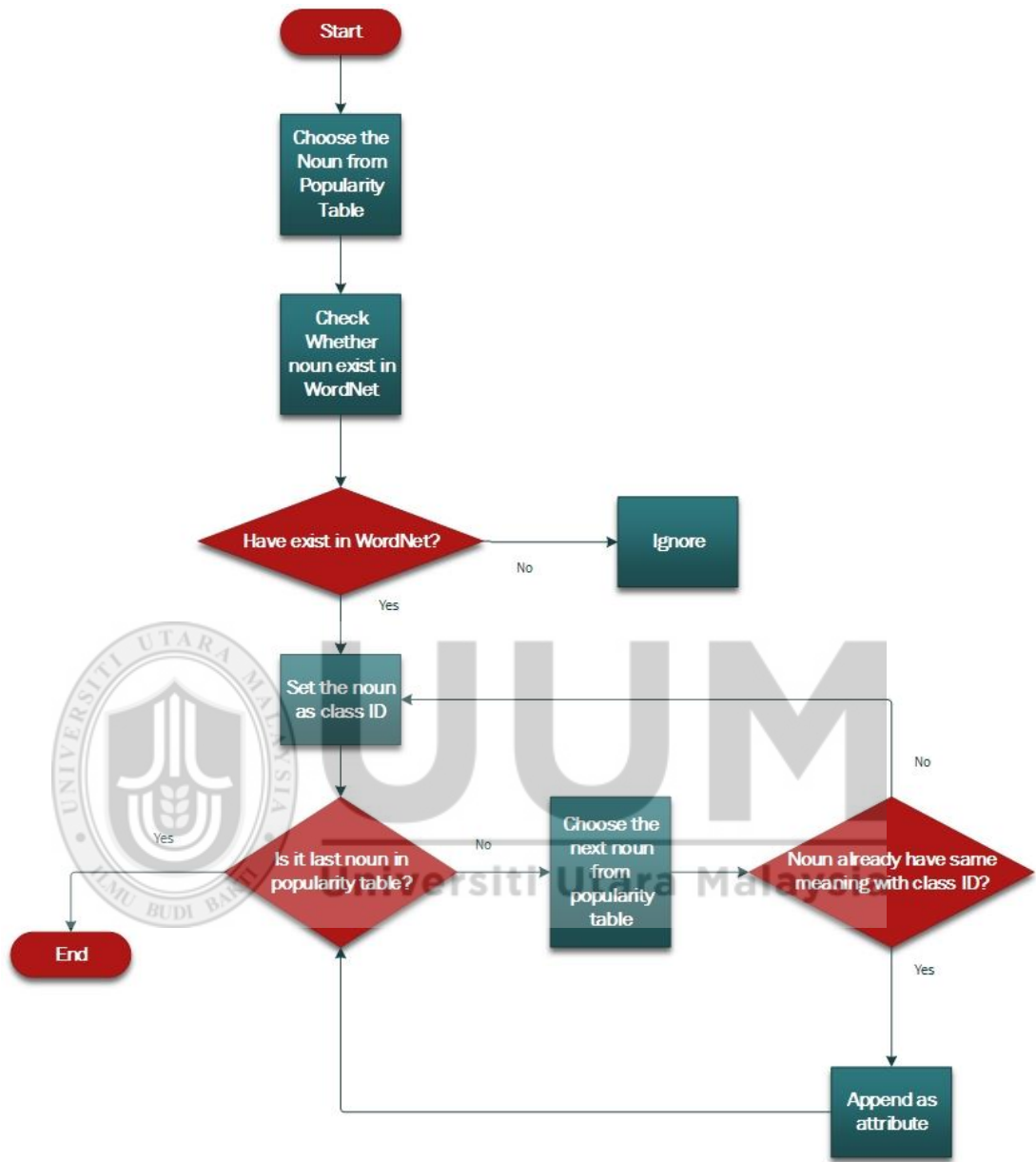


Figure 4.10: Flowchart of Classes Constructing

This step was performed by using the synset similarity program. It was implemented as part of the *wordnet* sub-package, which was part of the nltk package. A synset is a set of synonyms that share a common meaning. Each synset is composed of one or more *wordnet* lemmas, which represent specific meanings within a given synset. Synset similarity is a score that denotes how close two given meanings are, based on a particular taxonomy of meanings. A portion of the classes construction result is shown in Figure 4.11 below.

```

Talk.religion.misc group
=====**** Classes ****=====
Class Number: 1
Class ID: [Synset('chemical_element.n.01')]
Class Attribute: ['re', 'xenon', 'au', 'ti', 'al', 'cd', 'sg', 'ers', 'oxygen', 'ho']
Class Number: 2
Class ID: [Synset('time_period.n.01')]
Class Attribute: ['years', 'night', 'week', 'afternoons', 'summer']
Class Number: 3
Class ID: [Synset('quality.n.01')]
Class Attribute: ['morality', 'capability', 'capabilities', 'complexity', 'nature']

```

Figure 4.11: Portion of the Classes Result

Once the construction process was completed, the classes are appeared as shown in Figure 4.11, and each class was assigned a unique identifier. The result can be demonstrated with three categories of class Number, class ID, class Attribute, which

appeared when the set of documents had been fully processed as shown in Figure above.

4.6.3 Finding the Relation:

To perform this phase of CDM construction, a number of processes had gathered to find the suitable relations between classes the steps below explained the process to find the relations.

- 1- Parse the document to get a proper structure of the sentence by taking each word and determining its structure from its constituent part, which aim to recognize a sentence and designation a grammatical structure of it by following a slandered English grammar rules. The output from this process will be structured sentence as shown in the example below :

(S (NP the (N priest)) (VP (v read) (NP the (N bible)))

- 2- Semantic analysis has used the output of the parsing process which is structured sentence to determine the logical expressions. The logical expressions done by manipulating the sentence grammatical structure from parsing output and formed it as pair of logical expression as shown in the example below:

x. (bible(x) & read (priest, x))

- 3- Check the verb.

- 4- If the verb belongs to popularity table then check the nouns before and after the verb. Else ignore the verb.
- 5- Check if both nouns belong to the constructed classes from previous step (classes constructions) then establish the relation based on the verb .Else ignore the relation

Figure 4.12 shows the flowchart of finding the relation process.



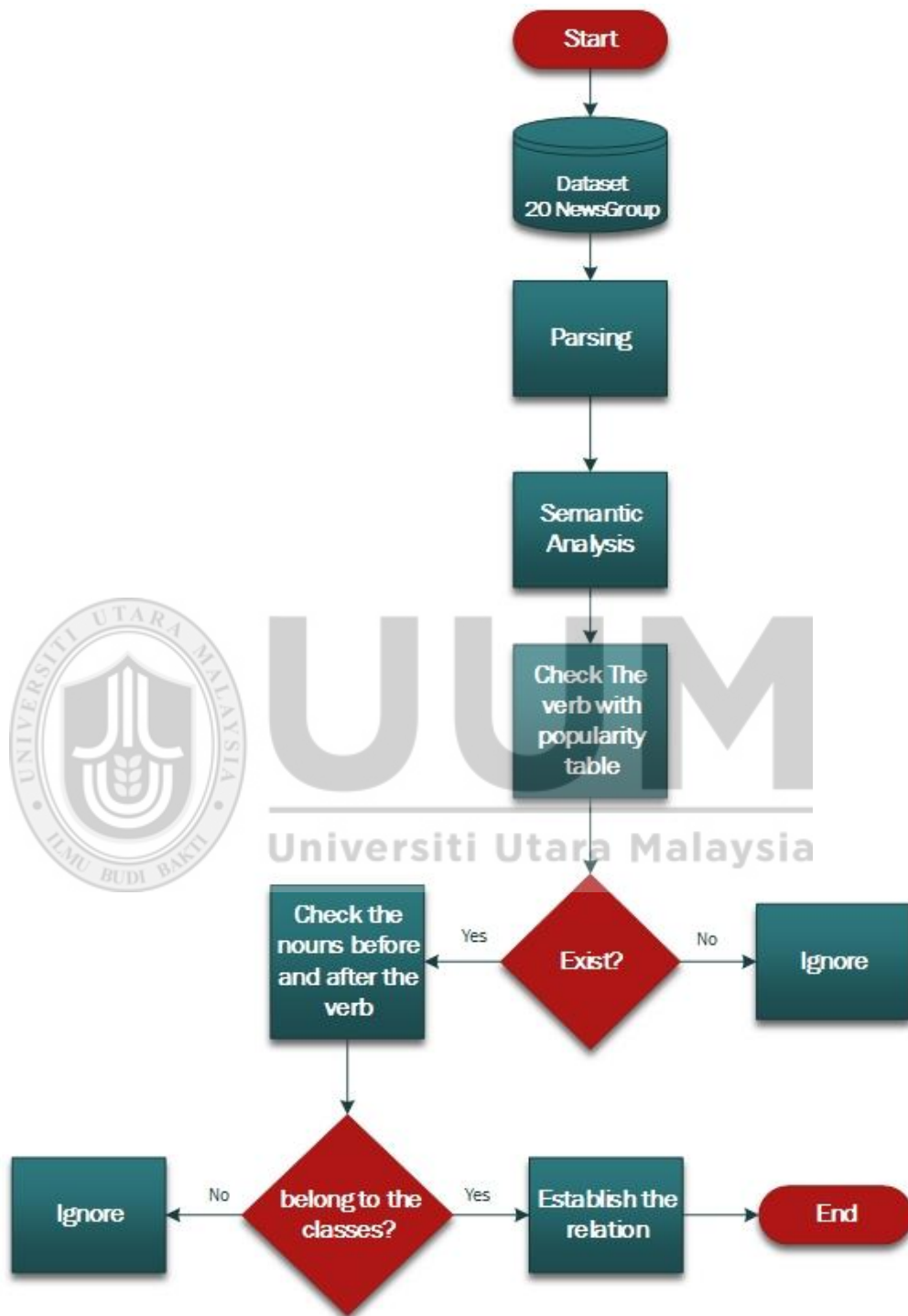


Figure 4.12: Flow Chart of Finding the Relation

Figure 4.13 shows a portion of the relationship between classes after perform the steps above.

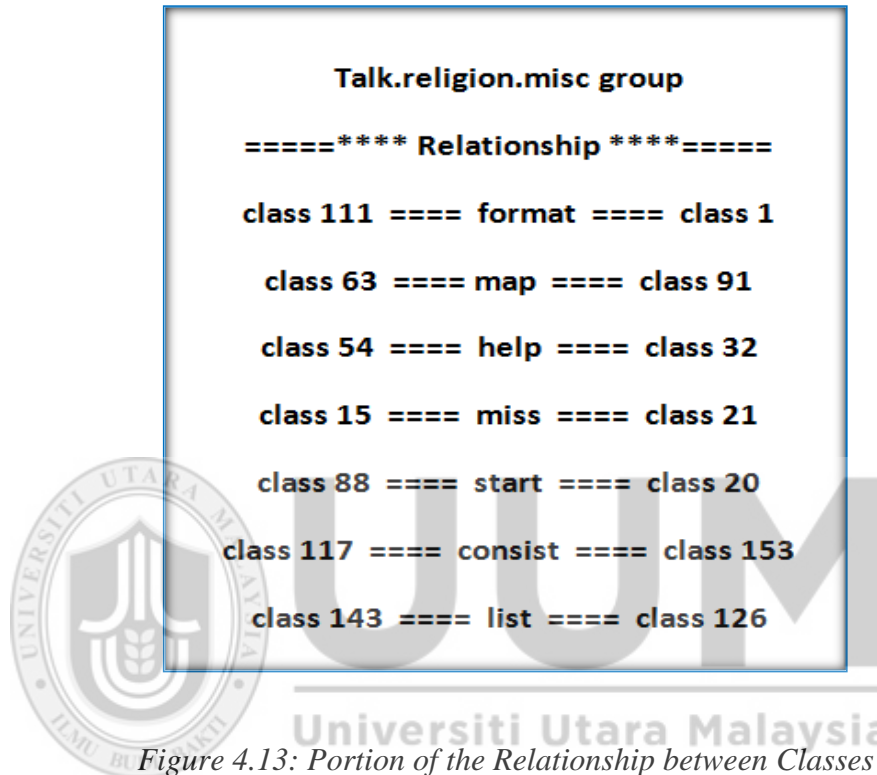


Figure 4.13: Portion of the Relationship between Classes

D: Drawing the graph:

Graph is a pictorial representation which includes a set of nodes and a set of edges. Using the result from the previous steps, the drawing was performed by using the classes and relationships between them. The steps below explain the way of drawing a graph and Figure 4.14 shows a Graphical Display for CDM.

1. The class_ID is used to represent the node, while each one of the class_ID represents its elements.

2. The relationship between the classes represents the connection between the nodes, and it represent as (\longrightarrow).

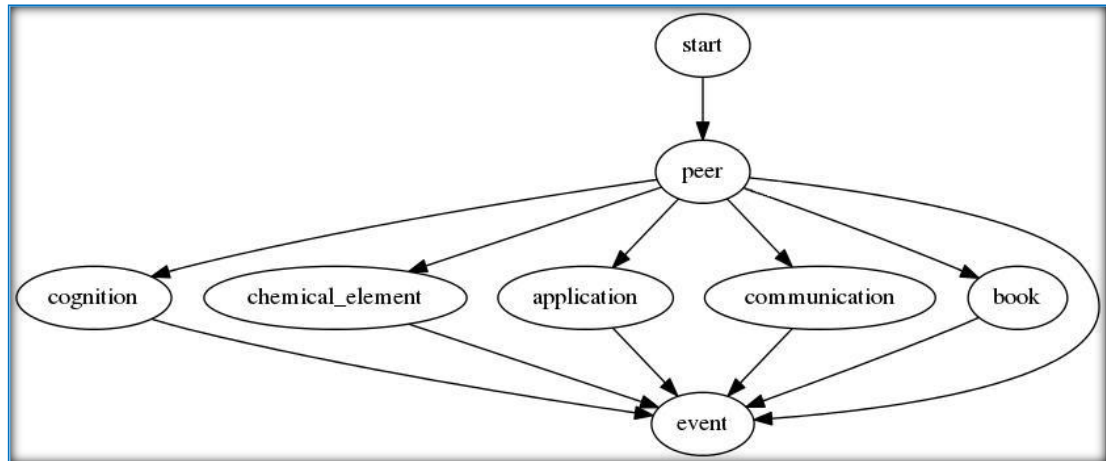


Figure 4.14: Graphical Display of CDM

Figure 4.14 shows the graphical display of *talk.religion.misc* CDM. It can be illustrate that each node represent one of the class_ID such as (book). Instead of represent the whole class (class_ID , elements), the class_ID only used through the phase of the graphical representation because each one of the class_ID represent its elements. The relationship between the classes represent as arrow to connect between the nodes.

The graph drawing step was implemented by running the python code and exported the class relationships to an external file named class.dat. This was an ordinary text file with a simple dependency structure. The external tool *makedot* converted this class information file into a DOT file in the *Graphviz* display language. Figure 4.15 shows a diagram for these steps and all the groups result are appended in **Appendix C**.

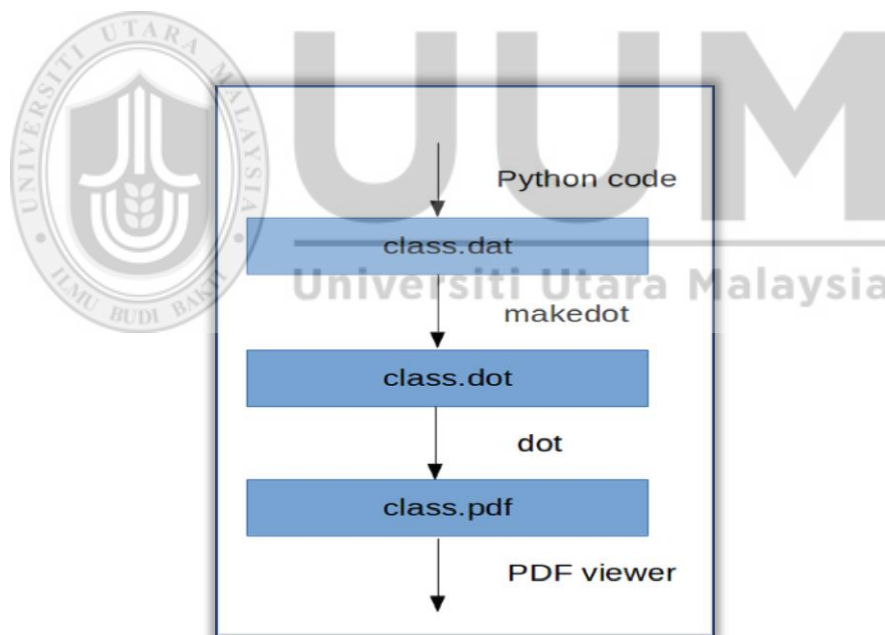


Figure 4.15: Diagram for CDM Model Creation

After completing all the phases required in CDM construction, the next step was to combine all processes involved in one flowchart to give a comprehensive view of a CDM as construction as shown in Figure 4.16.



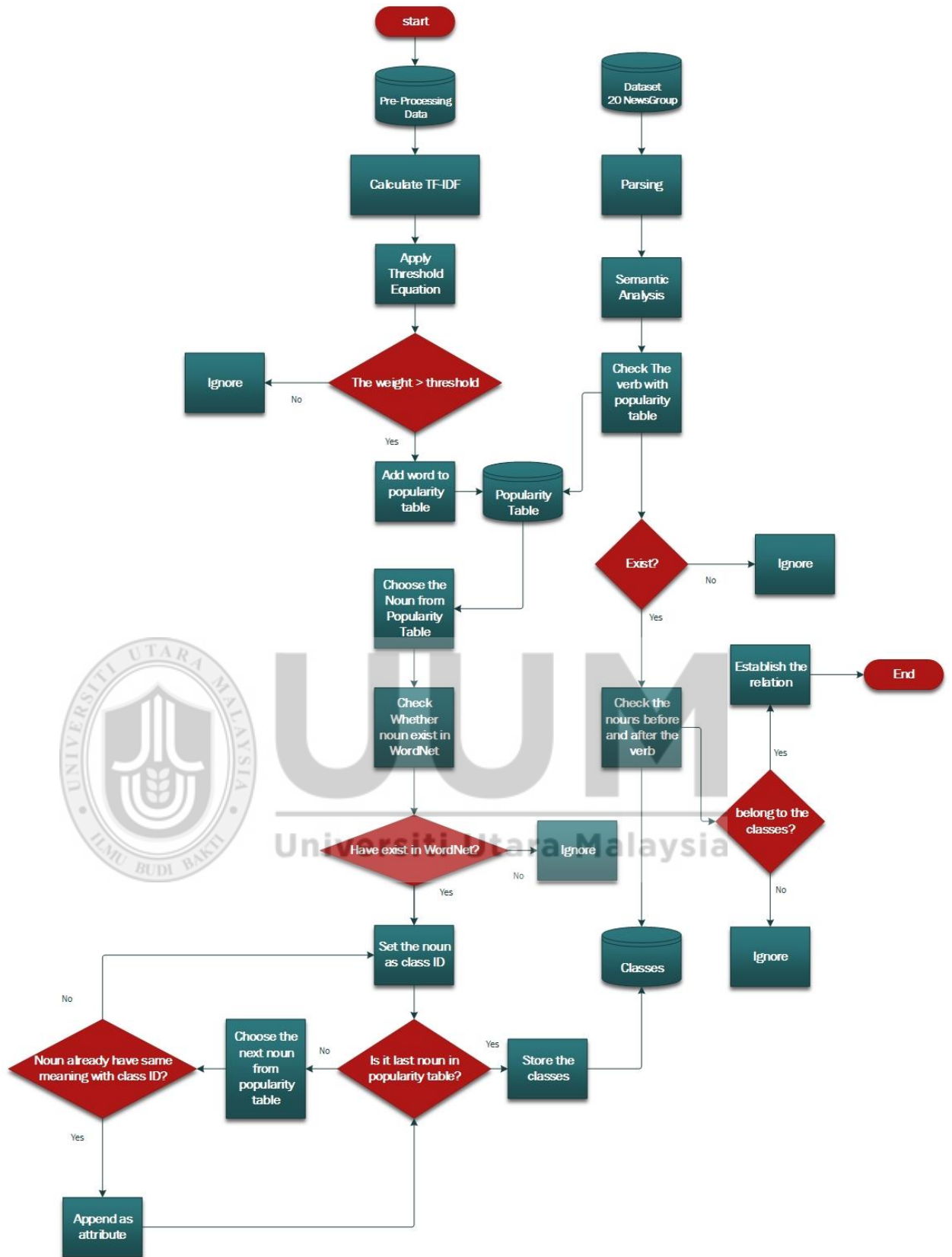


Figure 4.16: Flow Chart of constructing a CDM

In this research, the CDM was constructed for 20 different groups. Each group consisted of 1000 documents except *soc.religion.christian* group, which consisted of only 997 documents. All of the documents were used and the percentage of reduction words for each group were calculated before and after using the threshold. The reduction percentage can be referred in **Appendix D**.

For example, for **talk.religion.misc** group, the numbers of words before applying the threshold were 834. After applying the threshold, the number was reduced to 637 and this number were used to construct the CDM. Figure 4.17 shows the number of words before and after applying the threshold equation for **talk.religion.misc** group.

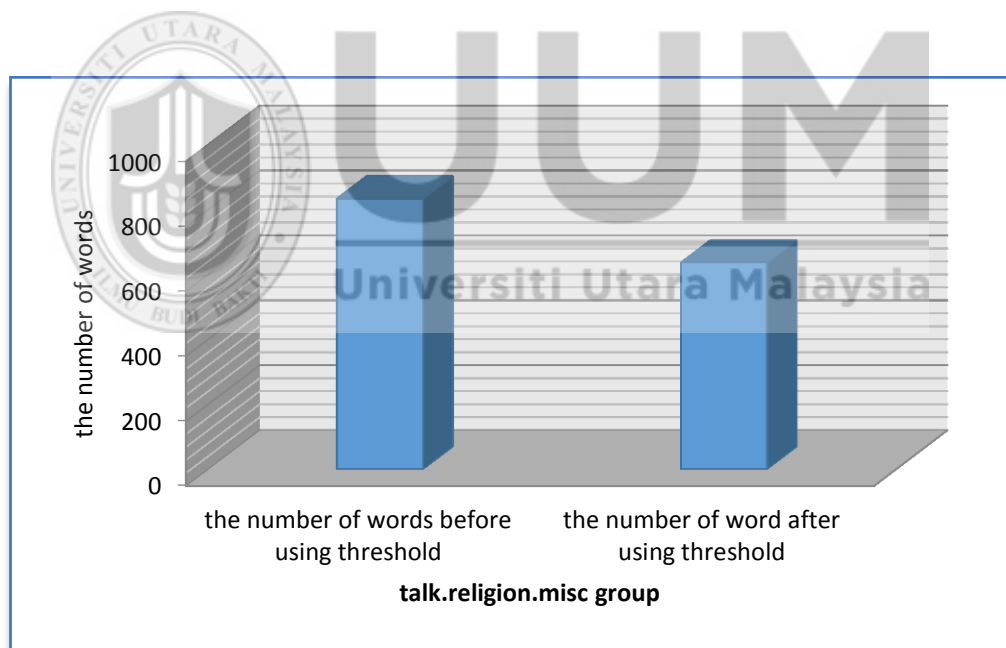


Figure 4.17: The Number of Words Before and After Using Threshold.

The reduction number of words was 197, which resulted from the calculation of subtraction between the number of words before and after using threshold. The percentage of reduction in this case was 24% and the percentage was calculated by dividing the number of the reduction words on the number of words before using threshold multiplied by 100. Figure 4.18 shows the percentage in words reduction for **talk.religion.misc** group.

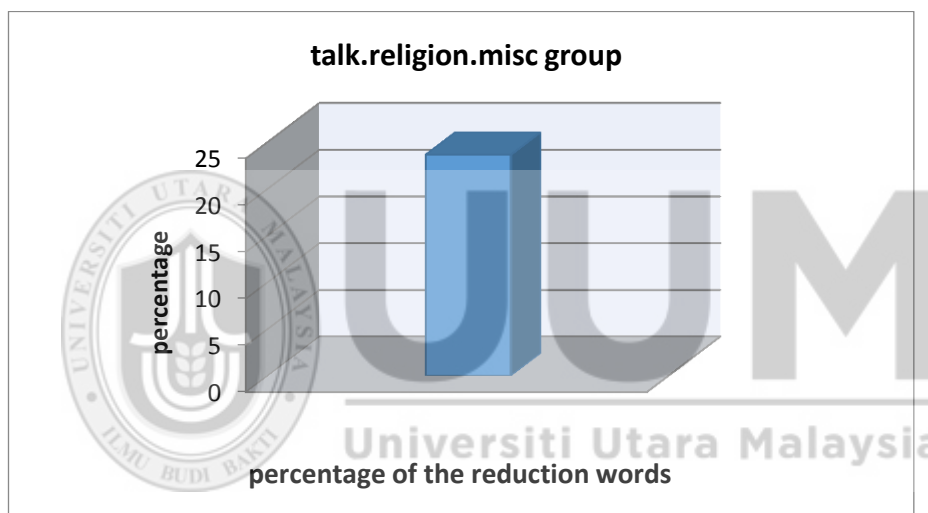


Figure 4.18: Percentage of Words Reduction

From this result it can be seen that the process of constructing a CDM was able to solve the sparsity problem by reducing the number of words and dealing with words that have higher weights than the threshold and ignored all the words under the threshold (Yuan, Chen, Xiang, & Wang, 2013). The graph in Figure 4.17 shows clearly the number of words before and after using the threshold. Threshold was

used to filter the useless words which have the low weight, and in turns handle the problem of sparsity by filtering the words.

CDM construction can also solve the semantic problem as well through building the meaningful classes, setting the suitable relations between them and discarding the meaningless words and the repeated words. For example, Figure 4.13 shows a portion of the relationship between classes for **talk.religion.misc** group. *Class 117* was connected to *Class 153* by the relationship called “consist”. The classes and their elements are shown below:

Class Number: 117

Class ID: book

Class Attribute: ['book', 'directory']

Class Number: 153

Class ID: communication

Class Attribute: ['print', 'section']

To prove that CDM also deal with semantic, some of the sentence to construct the classes above were traced back. The study found the original sentence for these classes were “*first five books consist different sections*” and “*muslims use the holy book which consist many sections*”

In the same figure, *Class 54* was connected to *Class 32*” by the relationship named “help” and below are the classes and their elements:

Class Number: 54

Class ID: science

Class Attribute: ['science', 'math', 'triangulation']

Class Number: 32

Class ID: group

Class Attribute: ['group', 'folk', 'collection', 'organization', 'band', 'data', 'class']

Also by tracing back some of the sentences to construct the above classes, the study found the original sentence for these classes were “*a bit of a reach science helps the organization to observe the fact “and” people do science to helps the group to understand the value of it “.*”

From the above classes and relations, it can be indicated that the construction of CDM can also solve the semantic problems through building the meaningful classes and setting the suitable relation between them.

4.7 Summary

This research has demonstrated a method that works on natural language text in order to produce class relationships over that text, as well as computing statistical quantities that are also of interest. The basic execution steps for the methodology of the research were discussed starting from the pre-processing step which consists of a number of processes until constructing a canonical data model. The complexity of the results was strongly dependent on the data subset used



CHAPTER FIVE

EVALUATION

5.1 Introduction

The previous chapter presents the detail implementations of text representation by using canonical data model. The manner by which the CDM solves sparsity and semantic problems was also investigated. This chapter presents the two types of model evaluation performed in this study namely: the first is face validity evaluation by the language experts and the second is the similarity measurement between the CDM's and the original data set.

5.2 Model Evaluation

Evaluation is the organized assessments to provide "useful feedback". This feedback helps to ensure the research objectives has been achieved and the evaluation provides information about whether the research need improvement or further future works. In this study, text representation by using CDM was evaluated based on two categories, the first is face validity while the second is Jaccard similarity.

Face validity is one of the most commonly used tests to measure model validity essentially. It is a test to measure what it is supposed to measure, the relationship between the validity of a model and its purpose is an important reason to deal with,

the notion of model validity discovered from its purpose. Once validity is seen as “usefulness with respect to some purpose”, then this naturally becomes part of a question, which involves the “usefulness of the purpose” itself. Thus, in reality, judging the validity of a model ultimately involves judging the validity of its purpose too (Barlas, 1996).

While, the similarity measure is also an important concept used to compare two objects and to determine whether they are related to the same topics. Similarity is associated with relevance, a similarity measure is employed to determine whether a document is likely to be relevant to a specified CDM. There are many different techniques to measures similarity such as (Dice coefficient, cosine coefficient and overlap coefficient). (J. Zhang & Rasmussen, 2001). It is an important aspect in general and validity of the results in a model-based study is crucially dependent on the validity of the model. An important reason has to do with the relationship between the validity of a model and its “purpose”. What matters is the aggregate output behavior of the model, the model is assessed to be valid if its output matches the “real” output within some specified range of accuracy, without any questioning of the validity of the individual relationships that exist in the model (Barlas, 1996). In this research, Jaccard coefficient was employed to measure similarity between test data set and CDM because it is widely used to assess similarity (J. Zhang & Rasmussen, 2001).

5.2.1 Face Validity

It is one of the techniques that used to validate the model (Sargent, 2005). Face validation has to deal with experts to evaluate the model and determine if it is correct and reasonable for its purpose. Experts are asked to make subjective judgments on whether the model result possesses sufficient for its intended purpose. To validate the model and the outcome of this model, this research employed a face validity technique. With regard to face validity, focus group technique was carried out with five English Language lecturers from the Language Centre in University Utara Malaysia. The focus group sessions used five guiding questions associated with validating the CDM model. Figure 5.1 illustrates the five sequence processes of the face validity technique.

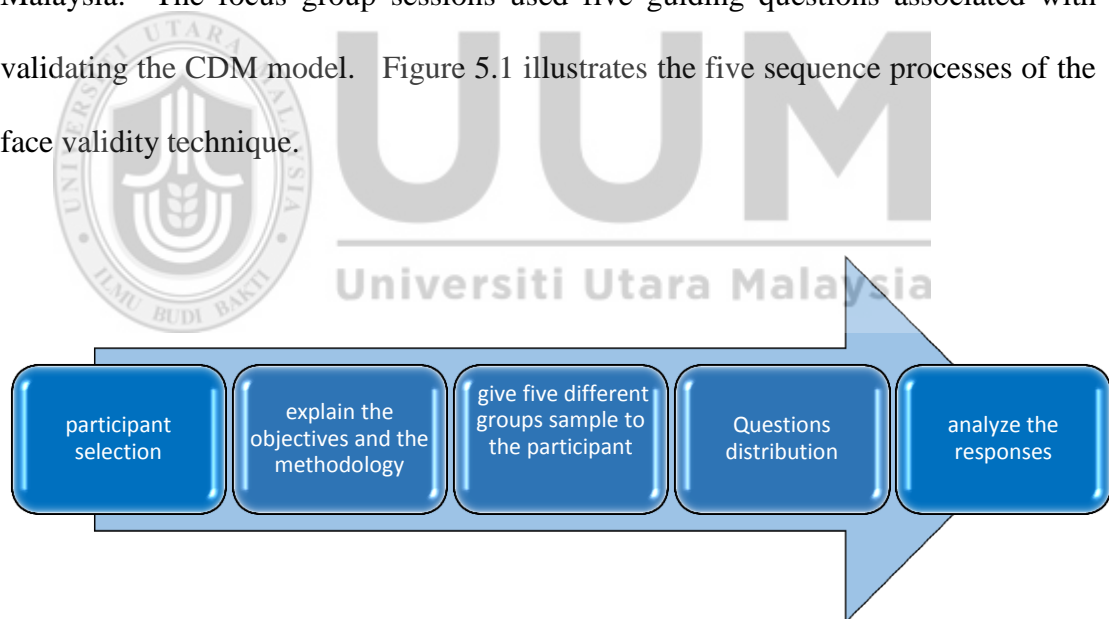


Figure 5.1 Face validity process

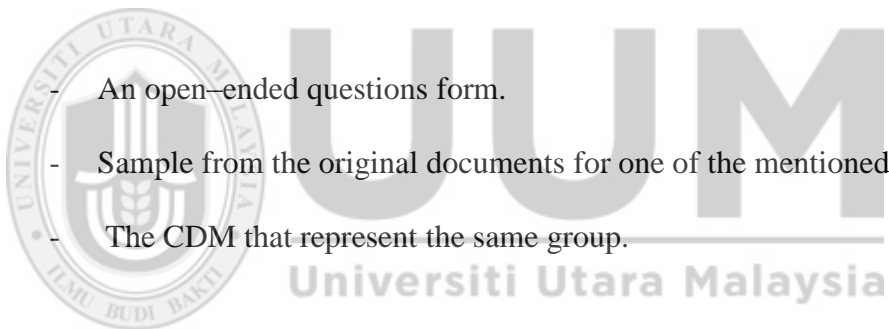
The selection of participants was based on a few strict criteria such as their qualification and experience in English language. The most important criteria for

choosing the participants was that they should have the formal university qualification in English language and they should have career related to English language. Another important criteria was that they have to have working experience in this English language field for at least three years and continuously still working during the sessions. Selection of participants was very important step because it influenced the whole process and the result of face validity. Table 5.1 shows the demographic of the face validity participants and the criteria.

Table 5.1 The Participants' Selection Criteria

participant	Gender	Place of work	Position	Qualification	Experience years
P1	Female	UUM	language instructor	MA applied linguistics	16
P2	Female	UUM	Language instructor	BA linguistic	20
P3	Female	UUM	Language instructor	B . English linguistic and literature	4
P4	Female	UUM	Language instructor	B. English linguistic and literature	3
P5	Male	UUM	Language instructor	B. Education (TESI)	3

Before conducting the focus group session, the problem and the objectives of research were clearly explained to all participants. The research method was also described to them as to give them full understanding about the processes and phases involved in the research. Five CDM were selected, and this includes [*alt.atheism*, *sci.electronics*, *misc.forsale*, *comp.os.ms-windows.misc*, *comp.graphics*] the order of these groups within the dataset was 1, 13, 7, 3, 2 were given to the participants to assess different groups instead of assessing the same group. This process was done to ensure that the model was able to represent different subjects correctly. Each participant received:

- 
- An open-ended questions form.
 - Sample from the original documents for one of the mentioned groups.
 - The CDM that represent the same group.

All of the participants were informed that they can freely give their honest answers. And also, all their personal information and responds will be used only for academic purpose.

According to (Barlas, 1996) there is an important relationship between the questions and the purpose of the model. (Creswell, 2013) said “Use open-ended questions without reference to the literature or theory unless otherwise dictated by the research design “

Therefore, the questions have designed to cover all the CDM construction phases and to guarantee that each phase was given a correct result. All the five questions were clearly clarified prior to the participants attempted to give their answers. Each one of the questions has been explained by showing example of one group called *talk.politics.mideast* by explaining terms used such as “what do we mean by class”, “class-ID”, “class elements”, “what is the relation“ and others. All the interview questions were appended in **Appendix F**.

Interestingly, almost all the experts agreed with the outputs of the model. They pointed out that the output was correctly related. The sections below summarize some of the participants’ answers based on the interview questions.

Question 1 “Do you find that our model can classify different text part correctly?”

Here is the answers:

Participant 1: “Yes. It obviously can “

Participant 2: “Yes, it defines every part correctly”

Participant 3:”some of them, at least to me“

Participant 4: “yes it can classify the different text part correctly “

Participant 5:”yes especially to teach English for example synonyms”

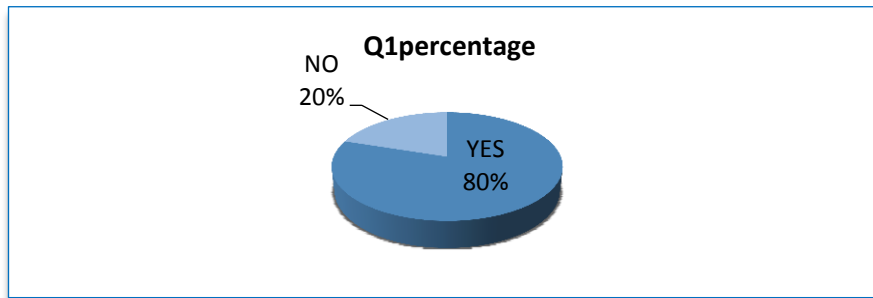


Figure 5.2 Percentage of response for Question 1

Figure 5.2 shows that the participants' responds for Question 1. It can be seen from the graph that there are 4 positive responds and only one negative respond which belong to participant 3. The percentage positive response for this question is 80%, which is highly positive.

Question 2 “Do you find the class_ID represent its elements?”, and the answers were as the following:

Participant 1: “Yes. ID represents synonyms “

Participant2:”the classification of groups represent its elements well .once we are well known about the subject and the progress “

Participant3: “from the printed pages given ,Yes”

Participant4:”Yes the class ID represent its elements”

Participant 5: “ Yes .Exactly “

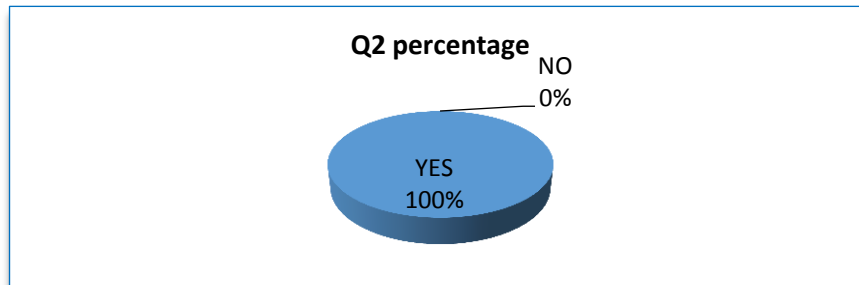


Figure 5.3 Responds percentage for Question 2

Figure 5.3 shows the participants responds for Question 2, where all the responds were positive and there is no negative respond for this question. The percentage is 100%.

Question 3 “Do you find the elements of our model are related correctly?” All the Participant agree that all classes elements are related correctly except Participant 2, and the answers were as the following:

Participant 1:”yes most of the words are inter-related”

Participant 2”I was not able to find the correlation“

Participant 3:”yes, from the presentation given “

Participant 4:”yes. They are correctly related”

Participant 5: "yes"

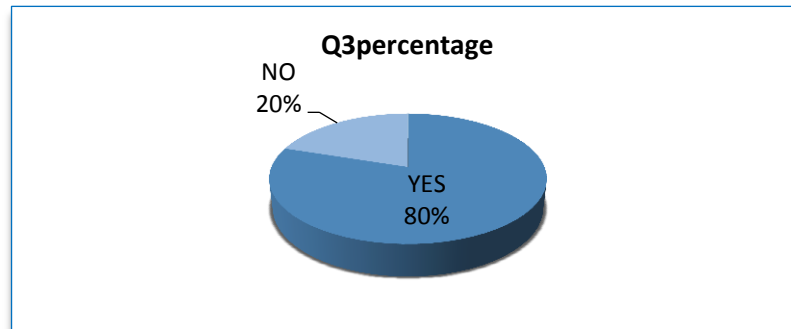


Figure 5.4 Responds Percentage for Question 3

Figure 5.4 illustrates the participants' responds for Question 3. It can be seen that there are 4 positive responds instead of one negative respond which belong to participant 2 and the percentage for this is 80% positive instead of 20% negative.

Question 4 "Do you find that the model can create the correct relation among the classes?" and the answers were:

Participant 1: "yes .they are grouped in such a manner that all words belong to the same group"

Participant 2: "it matches the group"

Participant 3: "Yes, I can see the correlation between the "property" and "happening" on pg5, for example..."

Participant 4: “Of course. I agree it can create the correct relation among classes”

Participant 5:”yes”

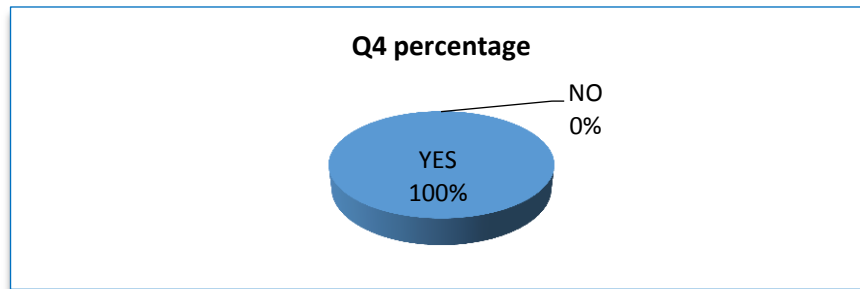


Figure 5.5 Responds Percentage for Question 4

Figure 5.5 presents the participants responds for Question 4, where it shows all the responds were positive and there is no negative respond for this question and positive response is 100%.

Question 5:”Do you think that the model results are understandable for human?

“and all the Participants denoted as :

Participant 1: “yes . but only those group that have basic knowledge in English grammar “

Participant 2:”yes. It is”

Participant 3:”yes .i think human can understand this”

Participant 4: "yes they are understandable for human "

Participant 5: " yes it is much more simple than using dictionary .programs makes it friendly and easier to use , also interesting. Good to interest technology in class "

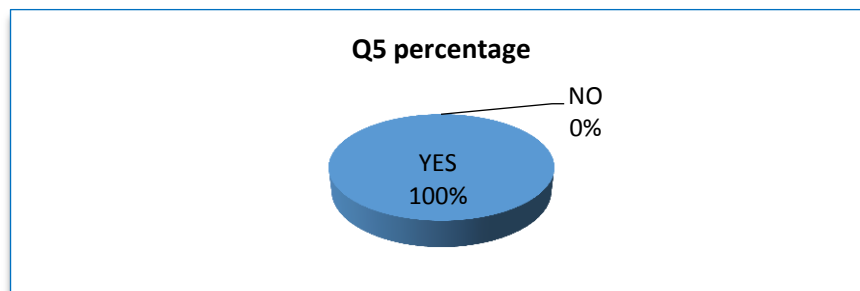


Figure 5.6 Responds Percentage for Question 5

Figure 5.6 explain the participants' responds for Question 5, all the responds were positive and there is no negative respond for this question and the percentage is 100%.

All of the participants' responses were demonstrated by "Yes" and "No" for all the questions to measure the percentage for each question, and they were reported accordingly as shown in Figure 5.7.

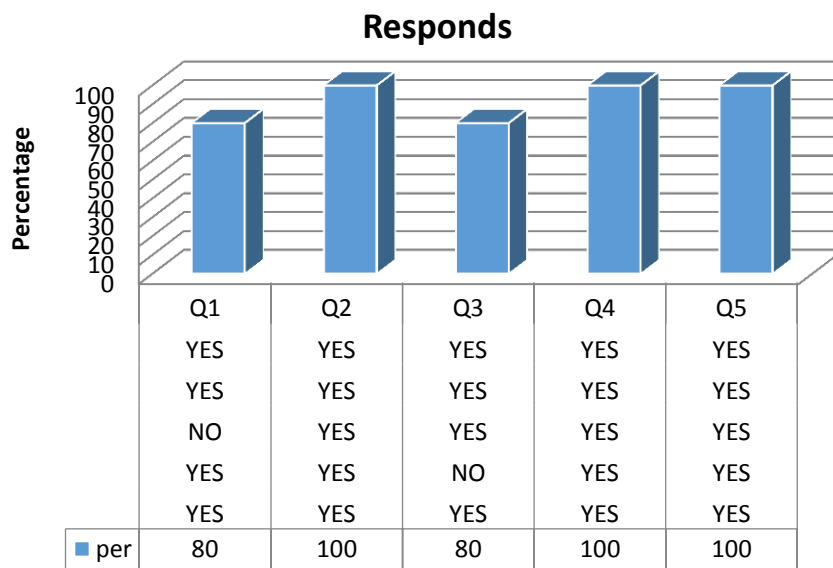


Figure 5.7 Participants Responds

From the above Figure, it can be concluded that out of 5 questions, 2 questions (Q1 and Q3) received 80% positive respond and 20% negative respond , while the rest questions (Q2, Q4 and Q5) gained 100% positive respond in score. Overall, the response's percentage for all questions was above 90%, which can be considered fairly well.

In conclusion, face validity approach used in this study can be considered as an acceptable technique for eliciting direct information from the participants to validate the model. The experts' opinion used as a technique to validate the model in light of the results obtained. The findings of this interviews session asserted that, the results from the model are logically acceptable as classes and relationship which indicated that the process of constructing CDM to represent text is valid.

5.2.2 Similarity Measure

Another method of evaluation in this study was Similarity Measure. This method reflects the degree of closeness or separation of the target objects (CDM) and the correspond data set. In other word, similarity measure maps the distance symbolic object into single numeric value which depends on two factors: characteristics of the objects and the measures itself. Moreover, choosing appropriate similarity measure is also crucial for evaluating the text representation.

In this research Jaccard similarity was selected to assess the similarity between the CDM and the test set (100 document had chosen from the original dataset). The objective was to measure the similarity as the intersection of the objects divided by the union of the same object. From this research perspectives, each group from the original dataset Composed of (1000) documents , the documents had divided into two set , first set contains (900) documents used to construct the CDM , second set includes (100) documents used to act as a test set to assess the similarity measurement. A finite number of documents from each group have compared with the CDM, from each group the (100) documents were selected randomly to work as a test set, by applying the Jaccard similarity in a *Rapidminer* software. The pre-processing step applied for each set were Tokenization process, removed all the stop word and stemming. Figure 5.8 shows the pre-processing step in rapid miner software.

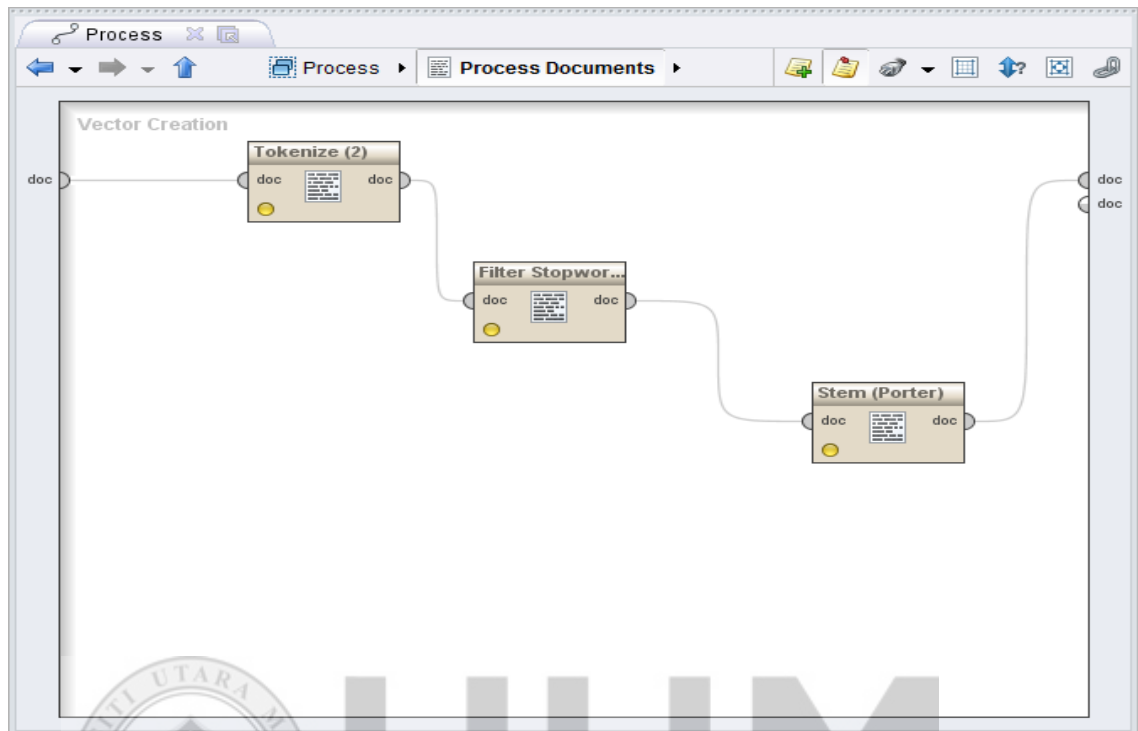


Figure 5.8: The Pre-Processing Step in Rapid Miner Software.

Figure 5.9 exhibits a screenshot from *RapidMiner* software main process, which shows 100 of documents from each group that have been compared with the 20 CDM by using *Jaccard Similarity*. As showed in the figure, there are 21 small blocks in the left side of the window which represent the 20 CDM plus one block contains 100 documents from one group. In the middle of the window, there is a big block represents the pre-processing steps. Finally, the single block in the right side of the window is the *Jaccard Similarity*. The 100 documents from each group were compared with 20 CDM for all groups in one single run. This process was repeated for 20 groups.

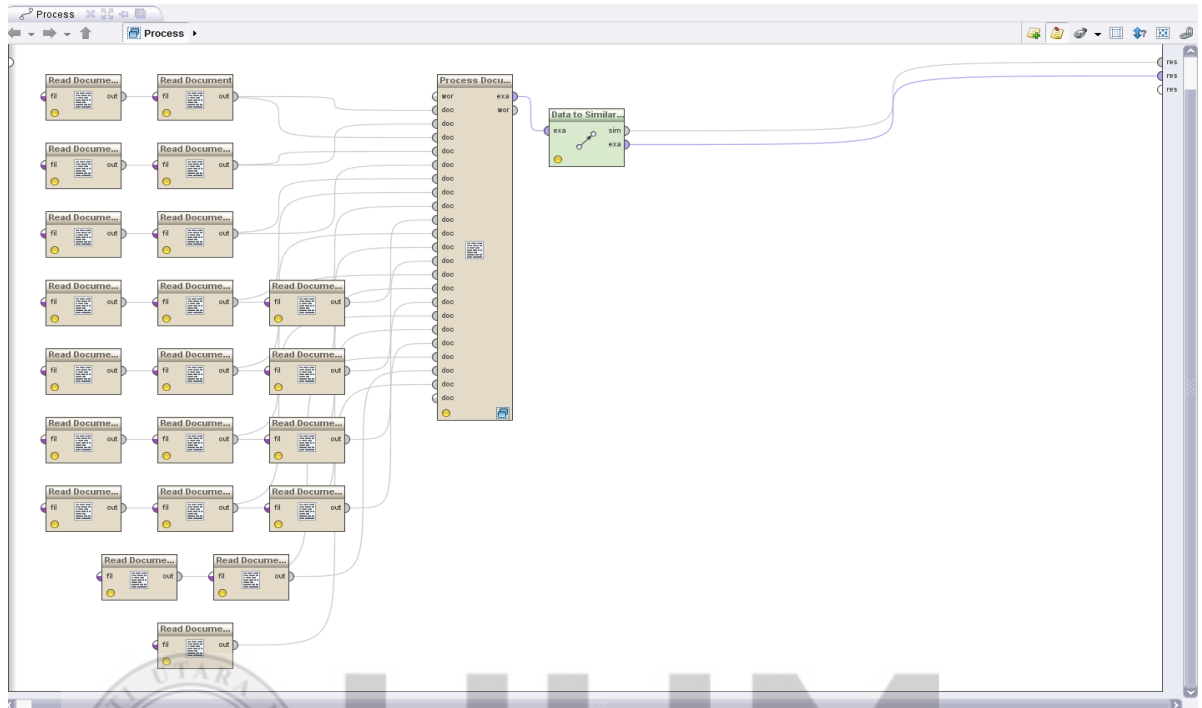


Figure 5.9: The Main Process.

After applying *Jaccard Similarity* measure for each group, the result was analyzed in MS Excel software and a graph was generated to each group as shown in the figures below. All of 20 similarity result graphs are attached in **Appendix G**. The x-axis represents the 20 CDM compared with 100 documents named in the middle of the graph. While, the Y-axis represents the similarity percentage.

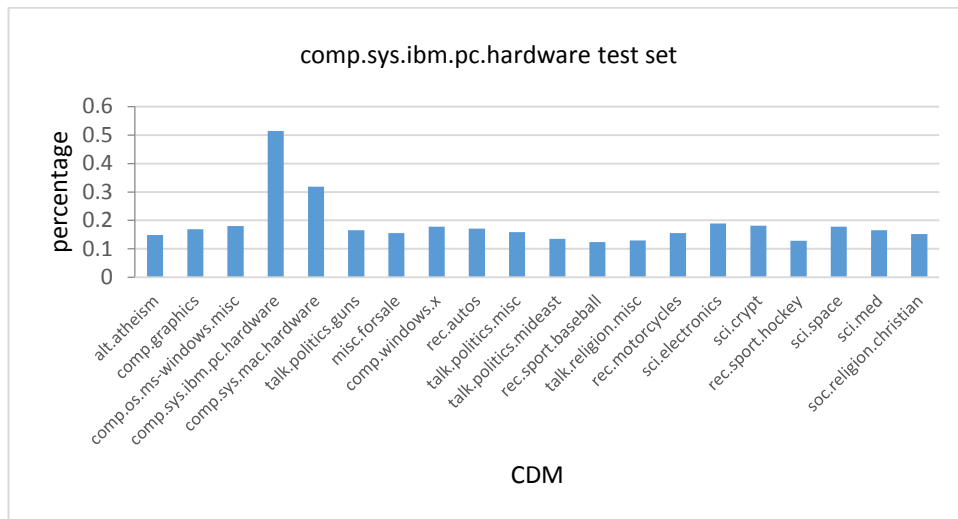


Figure 5.10: Similarity percentage of *comp.sys.ibm.pc.hardware* group

From the above figure, it can be concluded that each result file has high similarity percentage with the original text of the same group. Besides that, each result file has various percentages with the rest groups depending on how much these groups are related to each other. For example, Figure 5.10 shows the similarity result of *Comp.sys.ibm.pc.hardware* group, where this group of data set contains data about IBM hardware .which classified in the data set origins as part of computer category. From this figure, we can see that the result has high similarity percentage with the CDM of the same group. At the same time, the result from *Comp.sys.ibm.pc.hardware* group has also high percentage (32%) with CDM of *Comp.sys.mac.hardware* group due to both groups represent same subject which is computer hardware, so that both groups might have same words, classes and relations in CDM.

On the contrary, Figure 5.11 shows the different side of similarity result. The 100 documents from *Misc.Forsale* group have high similarity percentage with CDM of the same group. But, it has fairly low percentage with the CDM from the rest groups. The possible reason behind this result is may be the original group is not related or closed to any other group.

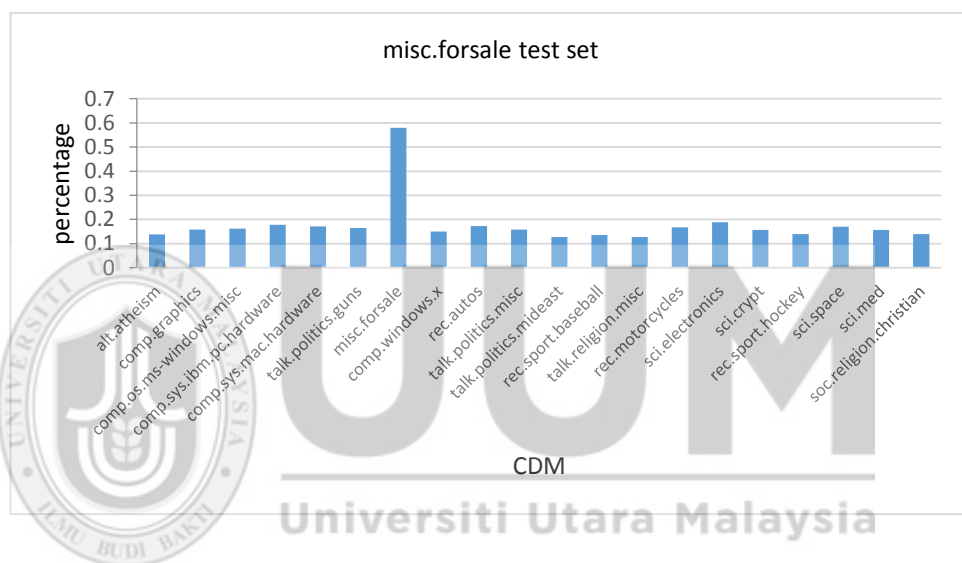


Figure 5.11: Similarity percentage of misc.forsale group

Table 5.2: Similarity Percentage for 20 CDM

CDM \ Dataset	alt.atheism	comp.graphics	comp.os.ms-windows.misc	comp.sys.ibm.pc.hardware	comp.sys.mac.hardware	talk.politics.guns	misc.forsale	comp.windows.x	rec.autos	talk.politics.misc	talk.politics.mideast	rec.sport.baseball	talk.religion.misc	rec.motorcycles	sci.electronics	sci.crypt	rec.sport.hockey	sci.space	sci.med	soc.religion.christian
alt.atheism	66%	15%	14%	15%	15%	22%	14%	13%	18%	24%	20%	17%	25%	18%	11%	18%	16%	16%	17%	23%
comp.graphics	14%	59%	19%	17%	19%	14%	16%	17%	15%	14%	10%	13%	12%	15%	13%	14%	13%	13%	12%	10%
comp.os.ms-windows.misc	13%	19%	62%	18%	20%	14%	16%	16%	16%	13%	10%	14%	12%	15%	13%	13%	13%	11%	12%	10%
comp.sys.ibm.pc.hardware	15%	21%	22%	51%	23%	16%	18%	17%	18%	15%	12%	15%	13%	17%	16%	15%	15%	14%	14%	11%
comp.sys.mac.hardware	13%	19%	19%	32%	59%	14%	17%	14%	17%	13%	10%	14%	11%	15%	14%	12%	13%	11%	12%	10%
talk.politics.guns	22%	17%	15%	17%	18%	57%	16%	14%	20%	24%	19%	18%	20%	20%	13%	19%	18%	17%	17%	18%
misc.forsale	13%	17%	16%	16%	18%	14%	58%	12%	16%	13%	11%	14%	11%	16%	13%	11%	13%	12%	12%	10%
comp.windows.x	15%	22%	19%	18%	19%	15%	15%	67%	16%	15%	12%	14%	13%	15%	13%	15%	13%	13%	13%	12%
rec.autos	17%	18%	17%	17%	19%	18%	17%	13%	60%	18%	14%	17%	15%	21%	15%	15%	16%	15%	15%	13%
talk.politics.misc	23%	17%	15%	16%	17%	24%	16%	14%	20%	54%	21%	18%	21%	20%	12%	19%	18%	17%	17%	19%
talk.politics.mideast	21%	14%	12%	14%	13%	22%	13%	11%	17%	23%	56%	15%	19%	17%	10%	16%	16%	15%	15%	18%
rec.sport.baseball	14%	14%	14%	12%	15%	14%	14%	10%	15%	14%	11%	52%	12%	15%	9%	10%	17%	10%	11%	10%
talk.religion.misc	23%	15%	13%	13%	14%	19%	13%	11%	16%	19%	16%	16%	52%	16%	9%	15%	14%	12%	14%	19%
rec.motorcycles	16%	17%	15%	16%	17%	18%	17%	12%	21%	17%	14%	17%	15%	59%	13%	14%	16%	14%	14%	13%
sci.electronics	15%	20%	19%	19%	21%	17%	19%	15%	20%	16%	13%	15%	14%	19%	63%	15%	15%	16%	15%	12%
sci.crypt	21%	20%	18%	18%	19%	22%	16%	17%	19%	22%	17%	17%	19%	18%	14%	56%	16%	18%	17%	17%
rec.sport.hockey	14%	14%	13%	13%	14%	15%	14%	10%	15%	15%	12%	18%	12%	16%	9%	11%	51%	11%	11%	11%
sci.space	19%	21%	16%	18%	18%	20%	17%	15%	19%	20%	16%	16%	16%	19%	15%	18%	16%	61%	18%	16%
sci.med	20%	18%	16%	17%	17%	19%	16%	14%	19%	20%	16%	17%	17%	19%	13%	17%	16%	17%	52%	16%
soc.religion.christian	25%	17%	14%	15%	15%	21%	14%	13%	17%	22%	19%	17%	24%	17%	11%	17%	16%	15%	17%	54%

As long as all the CDM's have high similarity percentage with its 100 original documents groups, using CDM for text representation has positive impact. The values of similarity percentages for all groups are gathered in Table 5.1. The highest percentage obtained the similarity between the CDM and the original data set. On other hands, the rest percentages represent how similar CDM with the original data from other groups.

The above table explains the similarity percentage where the columns represent the CDM model and the rows represent the 100 documents data set. The highest similarity percentage was for *comp.windows.x group* which is 67% while the lowest is for *comp.sys.ibm.pc.hardware group* is 51% and the rest of the groups were rated fluctuate between these two rates.

5.3 Results and Discussions

From the previous sections of 5.2.1 and 5.2.2, the research employed two techniques namely: face validity and similarity measure to evaluate the results. Each technique plays a vital role during the evaluation process to grantee that the research has correctly achieved the intended objectives.

Face validity is one of the most common methods to evaluate the model. Each model has different purposes, where the relationships between the model and its purpose is very important and become substantial when preparing guiding

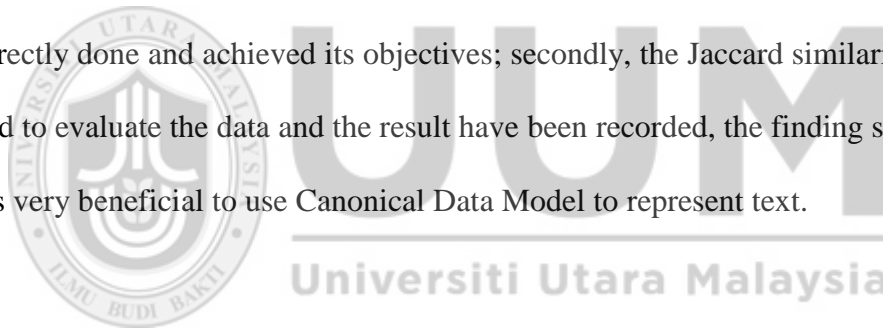
questions to ensure the right information gathered from the participants. The result of face validity was obtained by asking knowledgeable persons or experts in English language, whom use the English language as main part of their carriers. Each guiding questions were arranged to represent a particular phase in CDM construction to indicate that each one of phases had given the right result. The response from the participants were analyzed to prove that each given phases were the right one and logically acceptable. The overall analysis of face validity validated the model in this research.

In addition to face validity result, the CDM result had been tested also by applying the Jaccard similarity to measure the closeness and the separation of the CDM and the corresponding data set. The result of the measurement between CDM and the test was listed in Table 5.2 for all CDM. By surveying some literature it can be deduced that there are wide range of similarity functions and various authors have used them differently in the different domains. This research used Jaccard similarity as a part of evaluation phase (Truong, Amblard, Gaudou, & Sibertin-Blanc, 2014). From the result in table 5.2, group *comp.windows.x group* has the The highest similarity percentage (67%) while the lowest percentage was for *comp.sys.ibm.pc.hardware group* (51%) and the average of similarity for all groups was (57%) for all CDM .

Finally, based on the above result, it can be concluded that using CDM to represent different groups of data set was suitable but requires more investigation, and it can be utilized to reduce sparsity and semantic problems.

5.4 Summary

In this chapter, text representation was implemented by using Canonical Data Model. The result of the implementations were evaluated by two techniques: firstly, the result of the model was validated by the language experts from the Language Centre in UUM, whom concluded that the processes and the configurations of the CDM was correctly done and achieved its objectives; secondly, the Jaccard similarity have been used to evaluate the data and the result have been recorded, the finding showed that it was very beneficial to use Canonical Data Model to represent text.



CHAPTER SIX

CONCLUSION AND FUTURE WORK

6.1 Introduction

The previous chapters have presented the whole body of the research starting from identification of the problem and exploring the theoretical background, and the implementation of text representation by using Canonical Data Model. This chapter concludes the fundamental purpose of the research which is to reduce the sparsity and semantic problems that are related to the traditional text representation methods such as TF-IDF algorithm. Other part of this chapter provides details for the future research in this area.

6.2 Research Summary

This research investigated the current issues associated with sparsity and semantic for text representation technique, where the research addressed a number of weaknesses of the traditional algorithms to represent text. The essential contribution to this research lied in adapting and constructing the Canonical Data Model to represent text and reduced the sparsity and semantic problems which are associated with text representation methods.

A method is presented to construct canonical data model, where the construction is based on the needs to achieve the research objectives. The constructing was based on the needs to achieve the research objectives, the method of constructing benefit from canonical main Characteristics by integrating a set of documents into one comprehensive representation. The research achieved the integration in each data set by processing each group as one document instead of many documents. The proposed CDM was based on the utilization of method that works on natural language text in order to produce class relationships over that text.

The model was tested on 20 news data set and some of these data set are related to the others. The research demonstrated the effectiveness of using canonical data model to represent text based on the result of face validity and similarity measurement, which indicated that each CDM was similar to its original group.

6.3 Limitation

Although this research provided an effective model to represent text, there are still some limitations relating to data processing and classification. This research was limited to one data source in which content was categorized into certain domains.

6.4 Future Works

Based on the limitations mentioned, the researcher addresses certain issues that can be tackled in the further studies for the sake of enhancing the current CDM, including:

1. Deal with more than 2 types of part of speech, instead of dealing with verb and noun only.
2. Test the model with more complex languages such as Arabic.



REFERENCES

- Abdullah, A., & Fadhil, O. (2014). Implementation of Preprocessing Techniques in Datamining.
- Aggarwal, C. C., & Zhai, C. (2012). A survey of text clustering algorithms. In *Mining Text Data* (pp. 77–128). Springer.
- Agnihotri, D., Verma, K., & Tripathi, P. (2014). Pattern and Cluster Mining on Text Data. In *Communication Systems and Network Technologies (CSNT), forth International Conference on communication and network technologies 2014* (pp. 428–432).
- Alfawareh, H. M., & Jusoh, S. (2013). Resolving ambiguous preposition phrase for text mining applications. In *Computer Applications Technology (ICCAT), 2013 International Conference on* (pp. 1–5).
- Andrews, S., & Polovina, S. (2011). A mapping from conceptual graphs to formal concept analysis. In *Conceptual Structures for Discovering Knowledge* (pp. 63–76). Springer.
- Aphinyanaphongs, Y., Fu, L. D., Li, Z., Peskin, E. R., Efstathiadis, E., Aliferis, C. F., & Statnikov, A. (2014). A comprehensive empirical comparison of modern supervised classification and feature selection methods for text categorization. *Journal of the Association for Information Science and Technology*, 65(10), 1964–1987.
- Balmas, F. (2004). Displaying dependence graphs: a hierarchical approach. *Journal of Software Maintenance and Evolution: Research and Practice*, 16(3), 151–185.
- Barlas, Y. (1996). Formal aspects of model validity and validation in system dynamics. *System Dynamics Review*, 12(3), 183–210.
- Basu, T., & Murthy, C. A. (2012). Effective text classification by a supervised feature selection approach. In *Data Mining Workshops (ICDMW), 12th International Conference on Data Mining Workshops 2012 . IEEE* (pp. 918–925).
- Beck, F., & Diehl, S. (2013). On the impact of software evolution on software clustering. *Empirical Software Engineering*, 18(5), 970–1004.
- Bhaisare, R., & Nayyar, V. (2014). Analysis of Effective Pattern Discovery with Text Mining in Business Based Application. *International Journal of Research*, 1(11), 288–292.
- Bloechle, J.-L., Rigamonti, M., Hadjar, K., Lalanne, D., & Ingold, R. (2006). XCDF: a canonical and structured document format. In *Document Analysis Systems VII* (pp. 141–152). Springer.
- Bloehdorn, S., Cimiano, P., Hotho, A., & Staab, S. (2005). An Ontology-based Framework for Text Mining. In *LDV Forum* (Vol. 20, pp. 87–112).

- Boroch, N., & Heger, T. (2010). Vehicle environment description and interpretation using conceptual graphs. In *Intelligent Transportation Systems (ITSC), 13th International Conference on Intelligent Transportation Systems 2010 .IEEE* (pp. 1233–1236).
- Boyer, K. L., Martínez-Trinidad, J. F., & Carrasco-Ochoa, J. A. (2014). Editorial: Introduction to the special issue on supervised and unsupervised classification techniques and their applications. *Pattern Recognition Letters*, *41*, 1–2.
- Capasso, P. (2008). *BioInView: a system for integrating heterogeneous and distributed biological data sources*. University degli Studi di Napoli Federico II.
- Chekima, K., On, C. K., Alfred, R., Soon, G. K., & Anthony, P. (2012). Document categorizer agent based on ACM hierarchy. In *Control System, Computing and Engineering (ICCSCE), International Conference on Control System, Computing and Engineering 2012. IEEE* (pp. 386–391).
- Chen, M., Weinberger, K. Q., Sha, F., & Others. (2013). An alternative text representation to TF-IDF and Bag-of-Words. *arXiv Preprint arXiv:1301.6770*.
- Chen, X., & Wu, C. (2012). A Text Representation Method Based on Harmonic Series. In *IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2012* (pp. 1830–1834).
- Chen, Y., Sun, Y., & Han, B.-Q. (2015). Improving Classification of Protein Interaction Articles Using Context Similarity-Based Feature Selection. *BioMed Research International*.
- Christopher, D. M., Prabhakar, R., & Hinrich, S. (2008). Introduction to information retrieval. *Christopher. Cambridge Univ. Press. 2008*, p(544).
- Ciesielski, M., Jabir, A. M., & Pradhan, D. K. (n.d.). Canonical Graph-based Representations for Verification of Logic and Arithmetic Designs. *University of Massachusetts, Amherst, MA(01003)*.
- Cohen, W. W. (1998). WHIRL: A Word-based Information Representation.
- Coldicott, P. A., & Lane, E. (2009). Standard Based Mapping of Industry Vertical Model to Legacy Environments. Google Patents.
- Comer, E. (2010). *Canonical Data Model Design Guidelines*.
- Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications.
- Darwiche, A. (2011). SDD: A new canonical representation of propositional knowledge bases. In *IJCAI Proceedings International Joint Conference on Artificial Intelligence (Vol. 22, p. (819))*.

- Dayananda, P., & Shettar, R. (2011). Survey on Information Retrieval in Semi Structured Data. *International Journal of Computer Applications*, 32(8), 1–5.
- Delugach, H., & Lampkin, B. (2000). Troika: using grids, lattices and graphs in knowledge acquisition. *Working with Conceptual Structures: Contributions to ICCS*, 201–214.
- Deshmukh, S. N., Deshmukh, R. R., & Deshmukh, S. N. (2014). Performance Analysis of Different Sentence Oddity Measures Applied on Google and Google News Repository for Detection of Substitution. *International Refereed Journal of Engineering and Science (IRJES)*, 3(3), 20–25.
- Dey, A., & Prukayastha, B. S. (2013). Named Entity Recognition using Gazetteer Method and N-gram Technique for an Inflectional Language: A Hybrid Approach. *International Journal of Computer Applications*, 84(9), 31–35.
- Dietrich, J., Yakovlev, V., McCartin, C., Jenson, G., & Duchrow, M. (2008). Cluster analysis of Java dependency graphs. In *Proceedings of the 4th ACM symposium on Software visualization* (pp. 91–94).
- Dietrich, M., & Lemcke, J. (2011). A Refined Canonical Data Model for Multi-schema Integration and Mapping. In *IEEE 8th International Conference on e-Business Engineering (ICEBE), 2011* (pp. 105–110).
- Dietrich, M., Weissmann, D., Rech, J., & Stuhec, G. (2010). Multilingual extraction and mapping of dictionary entry names in business schema integration. In *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services* (pp. 863–866).
- Dolamic, L., & Savoy, J. (2009). Indexing and stemming approaches for the Czech language. *Information Processing & Management*, 45(6), 714–720.
- Durugkar, S. (2013). Various Issues in Implementing Cross Language Information Retrieval and Enhancing the Efficiency of Meta Search Tool. *International Journal of Emerging Technology and Advanced Engineering*, 3(2), 605–609.
- El-Shishiny, H., & Volkov, P. (2011). Systems and methods for building an electronic dictionary of multi-word names and for performing fuzzy searches in the dictionary. Google Patents.
- Fares, M., Oopen, S., & Zhang, Y. (2013). Machine learning for high-quality tokenization replicating variable tokenization schemes. In *Computational Linguistics and Intelligent Text Processing* (pp. 231–244). Springer.
- Fleuren, W. W. M., & Alkema, W. (2015). Application of text mining in the biomedical domain. *Methods*, 74, 97–106.
- Francis, M., & KN, R. N. (2014). An Algorithm for Plagiarism Detection in

- Malayalam Language Documents Using Modified n-gram Model. *National Conference on Indian Language Computing (NCILC 2014)*, (13), 10–15.
- Galescu, L., & Allen, J. F. (2001). Bi-directional conversion between graphemes and phonemes using a joint n-gram model. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*.
- Gardner, S. (2007). Ontology-based information management system and method. Google Patents.
- Gharehchopogh, F. S., & Khalifelu, Z. A. (2011). Analysis and evaluation of unstructured data: text mining versus natural language processing. In *Application of Information and Communication Technologies (AICT), 5th International Conference on Communication Technologies 2011*. (pp. 1–4).
- Gonzalez, M. A., Martí, J. R., & Kruchten, P. (2011). A canonical data model for simulator interoperation in a collaborative system for disaster response simulation. In *Electrical and Computer Engineering (CCECE), 24th Canadian Conference on Electrical and Computer Engineering 2011*. (pp. 1519–1522).
- Guo, X., Xiang, Y., & Chen, Q. (2011). A vector space model approach to social relation extraction from text corpus. In *Fuzzy Systems and Knowledge Discovery (FSKD), Eighth International Conference on Fuzzy Systems and Knowledge Discovery 2011* (Vol. 3, pp. 1756–1759).
- Hakim, A. A., Erwin, A., Eng, K. I., Galinium, M., & Muliady, W. (2014). Automated document classification for news article in Bahasa Indonesia based on term frequency inverse document frequency (TF-IDF) approach. In *Information Technology and Electrical Engineering (ICITEE), 6th International Conference on Information Technology and Electrical Engineering 2014* (pp. 1–4).
- Harish, B. S., Guru, D. S., & Manjunath, S. (2010). Representation and classification of text documents: A brief review. *IJCA*, 110–119.
- Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1-2), 177–196.
- Hsiao, D. K., Neuhold, E. J., & Sacks-Davis, R. (2014). *Interoperable Database Systems (DS-5): Proceedings of the IFIP WG2. 6 Database Semantics Conference on Interoperable Database Systems (DS-5) Lorne, Victoria, Australia, 16-20 November, 1992*. Elsevier.
- Hsiao, D., Neuhold, E. J., & Sacks-Davis, R. (2014). Interoperability between database models. In *Interoperable Database Systems (DS-5): Proceedings of the IFIP WG2. 6 Database Semantics Conference on Interoperable Database Systems (DS-5) Lorne, Victoria, Australia, 16-20 November, 1992* (p. 101).

- Hu, X., & Liu, H. (2012). Text analytics in social media. In *Mining text data* (pp. 385–414). Springer.
- Huang, C.-R., Šimon, P., Hsieh, S.-K., & Prévot, L. (2007). Rethinking Chinese word segmentation: tokenization, character classification, or wordbreak identification. In *Proceedings of the 45th Annual Meeting of the Acl on Interactive Poster and Demonstration Sessions* (pp. 69–72).
- Huang, X., & Wu, Q. (2013). Micro-blog commercial word extraction based on improved tf-idf algorithm. In *TENCON 2013-2013 IEEE Conference (31194)* (pp. 1–5).
- Hull, D. A. (1996). Stemming algorithms: A case study for detailed evaluation. *JASIS*, 47(1), 70–84.
- Jhanji, D., & Garg, P. (2014). Text Mining. *International Journal Of Scientific Research And Education*, 2(08).
- Jiang, C., Coenen, F., Sanderson, R., & Zito, M. (2010). Text classification using graph mining-based feature extraction. *Knowledge-Based Systems*, 23(4), 302–308.
- Jusoh, S., & Alfawareh, H. M. (2012). Techniques, Applications and Challenging Issue in Text Mining. *IJCSI International Journal of Computer Science Issues*, 9(6), 431–436.
- Kamaruddin, S. S., Bakar, A. A., Hamdan, A. R., & Nor, F. M. (2008). Conceptual graph formalism for financial text representation. In *Information Technology, 2008. ITSIM 2008. International Symposium on* (Vol. 3, pp. 1–6).
- Karaa, A., & Ben, W. (2013). A NEW STEMMER TO IMPROVE INFORMATION RETRIEVAL. *International Journal of Network Security & Its Applications*, 5(4).
- Kaur, M., & Kaur, N. (2013). Web Document Clustering Approaches Using K-Means Algorithm. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(5), 861–864.
- Korde, V., & Mahender, C. N. (2012). Text classification and classifiers: A survey. *International Journal of Artificial Intelligence & Applications (IJAIA)*, 3(2), 85–99.
- Kumar, P., Khulbe, S., & Dhama, H. S. (2012). Computer Program for Counting the Part of Speeches, Text Narrations by using Secondary Data Algorithm Techniques. *International Journal of Computer Applications*, 51(11).
- Lan, M., Tan, C.-L., Low, H.-B., & Sung, S.-Y. (2005). A comprehensive

- comparative study on term weighting schemes for text categorization with support vector machines. In *Special interest tracks and posters of the 14th International Conference on World Wide Web* (pp. 1032–1033).
- Lazim, R. Y., & Kamaludin, A. (2013). Going Native: Indexing architecture of eXist-db_ An Open Source native XML database system. In *TENCON 2013-2013 IEEE Conference (31194)* (pp. 1–4).
- Liu, M., & Yang, J. (2012). An improvement of TFIDF weighting in text categorization. *International Proceedings of Computer Science and Information Technology*, 44–47.
- Ma, J., Xu, W., Sun, Y., Turban, E., Wang, S., & Liu, O. (2012). An ontology-based text-mining method to cluster proposals for research project selection. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 42(3), 784–790.
- Malarvizhi, M. P., & Mohana, M. S. (2014). A Survey on Various Candidate Generator Methods for Efficient String Transformation. *COMPUSOFT, An International Journal of Advanced Computer Technology*, 3(2), 524–528.
- Melnikov, M. P., & Vorobkalov, P. N. (2014). Retrieval of Drug-Drug Interactions Information from Biomedical Texts: Use of TF-IDF for Classification. In *Knowledge-Based Software Engineering* (pp. 593–602). Springer.
- Miguelanez, E., Brown, K. E., Lewis, R., Roberts, C., & Lane, D. M. (2008). Fault diagnosis of a train door system based on semantic knowledge representation. In *Railway Condition Monitoring, 4th IET International Conference on Railway Condition Monitoring 2008*. (pp. 1–6).
- Mitchell, B. S., & Mancoridis, S. (2010). Clustering module dependency graphs of software systems using the bunch tool.
- Nawab, R. M. A., Stevenson, M., & Clough, P. (2013). Comparing Medline citations using modified N-grams. *Journal of the American Medical Informatics Association*, amiajnl–2012.
- Novak, J. R., Minematsu, N., & Hirose, K. (2013). Failure transitions for joint n-gram models and G2p conversion. In *INTERSPEECH* (pp. 1821–1825).
- Palmer, D. D. (2010). Text preprocessing. *HANDBOOK OF*, 9.
- Pan, S., Wang, L., & Xia, G. (2012). Mining association rules from consumer product safety cases based on text classification. *JCIT: Journal of Convergence Information Technology*, 7(9), 422–430.
- Panda, G., Panda, B. S., & Giridhar, B. (2015). Text Data Mining with Different Comparisons. *International Journal of Computer Science and Mobile*

Computing, 4(2), 7–13.

- Pandit, S. (2008). *On a robust document classification approach using TF-IDF scheme with learned, context-sensitive semantics*.
- Patel, C., Hamou-Lhadj, A., & Rilling, J. (2009). Software clustering using dynamic analysis and static dependencies. In *13th European Conference on Software Maintenance and Reengineering, 2009. CSMR'09*. (pp. 27–36).
- Patil, S. P., & Saraf, R. N. (2013). Two-Step Approach for Acquiring Semantic Relations from Textual Web Content. *International Journal of Innovative in Engineering and Technology (IJJET)*, 2(4), 145–150.
- Pedrosa, G. V., & Traina, A. J. M. (2013). From Bag-of-Visual-Words to Bag-of-Visual-Phrases using n-Grams. In *Graphics, Patterns and Images (SIBGRAPI), 26th SIBGRAPI-Conference on Graphics, Patterns and Images 2013* (pp. 304–311).
- Qu, Q., Qiu, J., Sun, C., & Wang, Y. (2008). Graph-based knowledge representation model and pattern retrieval. In *Fuzzy Systems and Knowledge Discovery, 2008. FSKD'08. Fifth International Conference on Fuzzy Systems and Knowledge Discovery* (Vol. 5, pp. 541–545).
- Ramos, J. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*.
- Reshamwala, A., & Mahajan, S. (2015). Automating sequence dataset generating by using SeqGen. In *Communication, Information & Computing Technology (ICCICT), International Conference 2015* (pp. 1–5).
- Roebuck, K. (2012). *Enterprise Information Management (EIM): High-impact Strategies-What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Emereo Publishing.
- Sangwan, U. (2014). A Search Technique On Databases Based On User Expertise. *International Journal Publication In Academic Education And Research – Computer Science and Engineering*, 1(1), 1–4.
- Sargent, R. G. (2005). Verification and validation of simulation models. In *Proceedings of the 37th conference on Winter simulation* (pp. 130–143).
- Schedl, M. (2012). # nowplaying Madonna: a large-scale evaluation on estimating similarities between music artists and between movies from microblogs. *Information Retrieval*, 15(3-4), 183–217.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, 34(1), 1–47.

- Segond, F., Schiller, A., Grefenstette, G., & Chanod, J.-P. (1997). An experiment in semantic tagging using hidden markov model tagging. In *ACL/EACL workshop on automatic information extraction and building of lexical semantic resources for NLP applications*.
- Shannon, C. E. (1951). Prediction and entropy of printed English. *Bell System Technical Journal*, 30(1), 50–64.
- Shneiderman, B. (1992). *Designing the user interface: strategies for effective human-computer interaction* (Vol. 3). Addison-Wesley Reading, MA.
- Siti Mahfuzah, S. (2011). *Conceptual Design Model of Computerized Personal-Decision AID (CompDA)*. Universiti Utara Malaysia.
- Sonawane, S. S., & Kulkarni, P. A. (2014). Graph based Representation and Analysis of Text Document: A Survey of Techniques. *International Journal of Computer Applications*, 96(19), 1–8.
- Specia, L., & Motta, E. (2006). A hybrid approach for extracting semantic relations from texts. In *In. Proceedings of the 2nd Workshop on Ontology Learning and Population* (pp. 57–64).
- Stevens, R., Goble, C. A., & Bechhofer, S. (2000). Ontology-based knowledge representation for bioinformatics. *Briefings in Bioinformatics*, 1(4), 398–414.
- Suchanek, F. M., Ifrim, G., & Weikum, G. (2006). Combining linguistic and statistical analysis to extract relations from web documents. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 712–717).
- Suguna, S., & Gomathi, B. (2014). Comparison between Clustering Algorithms Based On Ontology Based Text Mining Techniques. *International Journal of Advanced Research in Computer Science*, 5(7).
- Sureka, V., & Punitha, S. C. (2012). Approaches to Ontology Based Algorithms for Clustering Text Documents. *International Journal of Circuit Theory and Applications* Sept-Oct.
- Szymański, J., & Duch, W. (2010). Representation of hypertext documents based on terms, links and text compressibility. In *Neural Information Processing. Theory and Algorithms* (pp. 282–289). Springer.
- Truong, M. T., Amblard, F., Gaudou, B., & Sibertin-Blanc, C. (2014). To Calibrate & Validate an Agent-Based Simulation Model-An Application of the Combination Framework of BI solution & Multi-agent platform.
- Wang, L., & Liu, X. (2008). A new model of evaluating concept similarity. *Knowledge-Based Systems*, 21(8), 842–846.

- Wang, Y., Ni, X., Sun, J.-T., Tong, Y., & Chen, Z. (2011). Representing document as dependency graph for document clustering. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (pp. 2177–2180).
- Wei, D., Zhang, C. H., & Zhu, X. N. (2014). Service Selection Based on Rule and Statistics Model in WoT Smart Home. In *Applied Mechanics and Materials* (Vol. 681, pp. 244–248).
- Wei, T., Lu, Y., Chang, H., Zhou, Q., & Bao, X. (2015). A semantic approach for text clustering using WordNet and lexical chains. *Expert Systems with Applications*, 42(4), 2264–2275.
- Wei, X., Xiaofei, X., Lei, S., Quanlong, L., & Hao, L. (2001). Business intelligence based group decision support system. In *International Conferences on Info-tech and Info-net, 2001. Proceedings. ICII 2001-Beijing. 2001* (Vol. 5, pp. 295–300).
- Yuan, B., Chen, Q., Xiang, Y., & Wang, X. (2013). Event Detection and Recommendation Based on Heterogeneous Information. In *Proceedings of the International Conference on Information Engineering and Applications (IEA) 2012* (pp. 407–416).
- Yuanyuan, L., & Jianhu, W. (2011). RESEARCH ON TEXT MINING. *American Journal of Engineering and Technology Research* Vol, 11(9).
- Yun-tao, Z., Ling, G., & Yong-cheng, W. (2005). An improved TF-IDF approach for text classification. *Journal of Zhejiang University Science A*, 6(1), 49–55.
- Zhang, G., & Odbal. (2012). Sentence alignment for web page text based on vector space model. In *International Conference on Computer Science and Information Processing (CSIP), 2012* (pp. 167–170).
- Zhang, J., & Rasmussen, E. M. (2001). Developing a new similarity measure from two different perspectives. *Information Processing & Management*, 37(2), 279–294.
- Zhou, L. (2010). An approach of semantic web service discovery. In *Communications and Mobile Computing (CMC), International Conference on Communications and Mobile Computing 2010* (Vol. 1, pp. 537–540).
- Zimmermann, T., & Nagappan, N. (2008). Predicting defects using network analysis on dependency graphs. In *Proceedings of the 30th International Conference on Software Engineering* (pp. 531–540).
- Zwietasch, T. (2014). *Detecting anomalies in system log files using machine learning techniques*. Institut für Softwaretechnologie.