N-GRAM BASED KEYWORD TOPIC MODELLING FOR CANADIAN LONGITUDINAL STUDY ON AGING SURVEY DATA

by

Dhivya Jayaramn

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
August 2018

Table of Contents

List of Tables

List of Figures

Abstract

Canadian Longitudinal Study on Aging (CLSA) is a study and platform funded by the Canadian Institute for Health Research (CIHR) which focuses on why some people age healthier while others do not. To understand this, the research team conducted a population-based study of older adults aged 45-85 across Canada. During the interview, participants were asked a question which focused on getting their opinion about what promotes healthy aging. The response to this question is plain unstructured text data. This thesis focuses on identifying various themes present in the responses with the help of a novel topic modelling algorithm which uses n-grams. The responses are short and informal making it challenging for text mining.

Traditional topic modelling algorithms like Latent Dirichlet Analysis (LDA) consider the corpus as a Bag-of-Word (BOW) model in which the order of the words in the document is not considered. It also does not consider the inter-document frequencies of the words. Intra-document frequency of a word in short document is invariably zero because the words do not repeat. When these models are applied to short documents, they usually suffer from data sparsity issues, and this seems to be the reason they did not work well with CLSA survey data, which includes short and noisy text documents.

Hence we propose a novel model using character n-grams which considers the relationship between the words in a document and inter-document frequencies of the words in the corpus. This solves the problem around noisy and sparse data and generates distinct topics. Experts evaluated the results produced by LDA, Token Keyword Model (TKM) and our proposed model and found that our model produced more distinct topics. We also calculated the intra topic distance and inter topic distance which showed that our model had the lowest intra topic distance and highest inter topic distance confirming that the topics do not overlap.

List of Abbreviations used

BTM - Biterm Topic Model
CHIR - Canadian Health Institute for Research
CKM - Character N-gram Keyword Model
CLSA - Canadian Longitudinal Study on Aging
CSV - Comma Separated Value
IoT - Internet of Things
LDA - Latent Dirichlet Allocation
LSA - Latent Semantic Analysis
NLP - Natural Language Processing
PLSA - Probabilistic Latent Semantic Analysis
SQL - Structured Query Language
SVD - Singular Value Decomposition
TKM - Token Keyword Model

Acknowledgements

I would like to thank each and everyone who helped and supported me throughout the last two years. First of all, I would like to profoundly thank my supervisor Dr. Vlado Keselj for his support and guidance throughout my time at Dalhousie. His insights helped me to take the right direction in my next step forward. I would like to extend my gratitude towards my co-supervisor Dr. Susan Kirkland for her time and advice from an epidemiological perspective which played a crucial role in shaping my thesis. I would like to thank my supervisors and CLSA Catalyst grant for funding me throughout my thesis. I would also like to thank my committee members Dr. Evangelos Milios and Dr. Srinivas Sampalli for taking their time and effort to review my thesis work on a short notice. I would also like to thank my research project group members for their guidance and direction when needed.

I would like to extend my indebtedness to my mother Shyamala, father Jayaraman and sister Haritha for supporting me emotionally without which I would not have accomplished this milestone in my career. I would like to thank all my friends who encouraged and pushed me harder to get where I am today. I would like to thank JeyaBalaji, Yamani, Dijana, Stacey and Magdalena who were always ready to give their inputs when needed and helped me fix issues with my codes. I would like to thank my roommates Pooja and Ruhi for always being there for me. Finally, I would like to extend my gratitude to my friends Deepika, Dwarakesh and Kumaran for providing a kind ear whenever needed. Without the support of anyone mentioned above, I would not have been able to complete this journey.

This research was made possible using the data/biospecimens collected by the Canadian Longitudinal Study on Aging (CLSA). Funding for the CLSA is provided by the Government of Canada through the Canadian Institutes of Health Research (CIHR) under grant reference: LSA 9447 and the Canada Foundation for Innovation. This research has been conducted using the CLSA dataset [Baseline Tracking Dataset version 3.2, Comprehensive Dataset version 3.1], under Application Number [170305]. The CLSA is led by Drs. Parminder Raina, Christina Wolfson and Susan Kirkland.

Chapter 1

Introduction

Researchers and organizations usually conduct surveys to collect information on their products or services. The survey helps them to understand how they can improve their product or service. The government also conducts a census survey on the population to collect data on demographics and socio-economic status of the people. The data collected through such surveys are enormous and often include unstructured text data. The comments do not contain any sentence structure which makes it difficult to process the data. It takes a lot of human effort to read or skim through all the data collected and identify different domains or topics which concerns the customers or users of the product or service. Once the domains are identified we can find the factors which are associated with the feedback or comments like the geographical region the customer resides, gender or any other factors.

Text mining techniques can help us to get useful information from these unstructured text data and to identify the various domains which can be improved to attract the customers. Generally, text mining techniques facilitate information retrieval, pattern recognition, classification and many other tasks. By using these techniques we can decipher various patterns or commonalities among the opinions given in the survey. We can also identify the features of the product or service which needs improvement or which does not make much sense to the user or a new feature which needs to be added for the convenience of the user.

1.1 Motivation

Data collected from social media or product review platforms do not answer to any specific question. They do not post on such platforms until they have a strong opinion about any event or product. But in a survey the participant is forced to answer a question even when they do not have any strong opinions which leads to generic answers. Retrieving information from such data using text mining techniques is challenging.

Aligning with the goals of Canadian Longitudinal Study on Aging (CLSA) which includes identifying the ways to age healthier, predicting diseases earlier, etc. finding the various opinions about healthy aging according to the elderly community in Canada plays a crucial role. For this purpose, CLSA conducted an interviewer-administered survey interview to study the factors of aging which can be used to help people in Canada and around the world. During the survey, the participants were asked a series of closed and open-ended questions. There was one particular question which focused on understanding what the elderly community thought about aging healthier. This question was open-ended, implying that the participants were not restricted to a set of pre-defined answers. Through open-ended questions, one can collect a whole lot of possible options and the participants are allowed to give their thoughts, feelings or concerns about the question. This open-ended nature of the question gathered a varied range of informal answers which were grammatically incorrect and conversational. Adding to this, the answers were recorded as the participants were speaking. This led to a considerable about of typographical errors in the collected data.

Most of the specific issues surrounding the CLSA data can be associated with any survey which contains open-ended questions. For surveys of this nature, it is important to understand the various themes or patterns in the answers collected. It becomes mundane to manually perform this task. It is also hard to implement machine learning or deep learning techniques due to the limited size of the data. It is much easier to employ text mining techniques like topic modelling to identify the themes. But the traditional topic modelling approaches fail to work with this type of survey data because of the short length of answers, typographical errors, informal structure of sentences.

1.2 Survey Data

According to the Prarie Research Associates [1], the history of survey dates back to the Middle Ages, where the Emperors used survey as a tool to understand the population characteristics and living standards of their people. Nowadays, surveys are designed carefully by researchers for a specific goal. Surveys are conducted for various purposes like research, to understand the consumer feedback on products and services or what people opine on an issue or policy.

Robert in his paper [2] states that the survey research can be separated into three eras. The period from 1930 to 1960 is known as the "Era of Invention". During this period methods which offered estimates and measurable bias-free sampling and error estimations became popular. It was then that US government was also interested in studying the social and economic status of its population. The second era was from 1960 to 1990 when modern information technology was introduced to people. Telephones and computers entered the world of the survey which increased the rate of responses and the number of surveys taken by the government also kept growing. The third era is from 1990 to the present, wherein mobile phones and the internet became popular. Surveys were able to reach people around the world, but there was also a drop in the response rate because it gave people an option to terminate the survey if they were no longer interested. But still, there were a lot of data collected which could be used for analysis because it reached a larger crowd. The traditional techniques did not work well but new techniques like text mining paved the way to survey data analysis or data analytics.

1.2.1 Survey Data Collection

There are many modes in which the survey data can be collected. Traditionally data was collected using paper-pencil or face-to-face interviews. The advent of technology provided various modes to collect data such as telephonic interview, web survey or mixed mode interviews. Each one has its own advantages and disadvantages.

Face to face interview is one of the traditional forms of collecting data. It is easy to gain trust and convince people to complete the survey. The interviewer can also understand the way the interviewee feels about certain sensitive issues with their

facial expression which is an added advantage. With the increased use of telephones and cell phones, organizations have started to collect data through telephone calls. Though the interviewer cannot connect to the interviewee emotionally because of location constraint, they can still convince the interviewee to complete the survey and understand their views. One major disadvantage of this mode is that the interviewee has the control to answer the call or not. But these interviews are time-consuming, costly and also can have limited reach for a fixed group of audience.

The most popular and easiest form of surveys is the web survey. Through the internet, it is easy to reach a bigger crowd but there is no guarantee that the survey is going to be taken or completed. This makes it error prone and leads to low response rate. Organizations also have started mixed-mode surveys, in which they conduct surveys in more than one mode. This makes the interviewee choose his or her most comfortable way to be interviewed.

1.2.2 Data Types and Storage

The data collected through surveys are of two types, quantitative and qualitative. Quantitative data are usually numbers or categories and can be represented in graphs or charts which are easily understandable. Statistics plays an important role in eliciting useful information from the data collected. Descriptive statistics are used to summarize or describe the information conveyed by the data. Insights obtained through this method applies only to that data. Whereas inferential statistics enable us to generalize the insights obtained from a dataset to a group of the population. Qualitative data are collected through an open-ended survey questionnaire. It generally contains images, text, videos or speech files. Such surveys are conducted on a limited or focused group of individuals. Here the participants are allowed to speak about what they feel or think about a particular area which results in data-driven research.

The datatype of the data collected through surveys is of three types, structured, unstructured and semi-structured data. In structured data, the responses or data models are predetermined. The user has to give answers which match the data model or select one among the given choices. Such data can be stored in tables or spreadsheets with titled rows and columns. When the dataset is large it is usually stored in a Relational Database [3]. If it is small enough it can be stored in a simple comma

separated value file (CSV). Such data are easy to process by using text mining algorithms as they have definite features which can be visualized into graphs or converted to data points for machine learning algorithms or use Structured Query Language (SQL) [3] queries to get the appropriate information. Usually, organizations store basic customer information in a structured format.

When the responses cannot be classified readily then they are known as unstructured data. Any text, image or videos can be classified as unstructured data. It includes data which are structured but still cannot be used to infer valid information. The main source of unstructured data includes social media, open-ended survey questions, Internet of Things (IoT) [4] applications etc. However, there are many challenges which need to be faced. By taking advantage of the various algorithms in Natural Language Processing (NLP) [5], Machine Learning [6], Deep Learning [7] and Big Data [8], we extract useful information from unstructured data. These data can be stored in NoSQL [9] database or as CSV files.

In certain surveys, the data collected can be a mixture of both structured and unstructured data. Such data are known as semi-structured data. These data may contain internal tags or marking which can be helpful in classifying or grouping the collected data. One good example of such type is email. By using the timestamp, subject or sender details, the emails can be grouped.

1.3 Text Mining

Unstructured data can have many underlying or hidden information which can be useful for research purpose or business insights. Such valuable and structured information can be extracted from massive unstructured data using Text Mining Algorithms. They help in converting the key content of the documents into quantitative data; i.e., convert free text into numbers by indexing them. It can also help in identifying the facts and the relationships between them. The main advantage of these methods is that we can make the algorithm understand similar words and context of the words in a sentence.

Text mining techniques can be used to classify or summarize the unstructured data collected through a survey. Through classification, we can identify the various domains underlying in the data. It is time-consuming for any human to read and

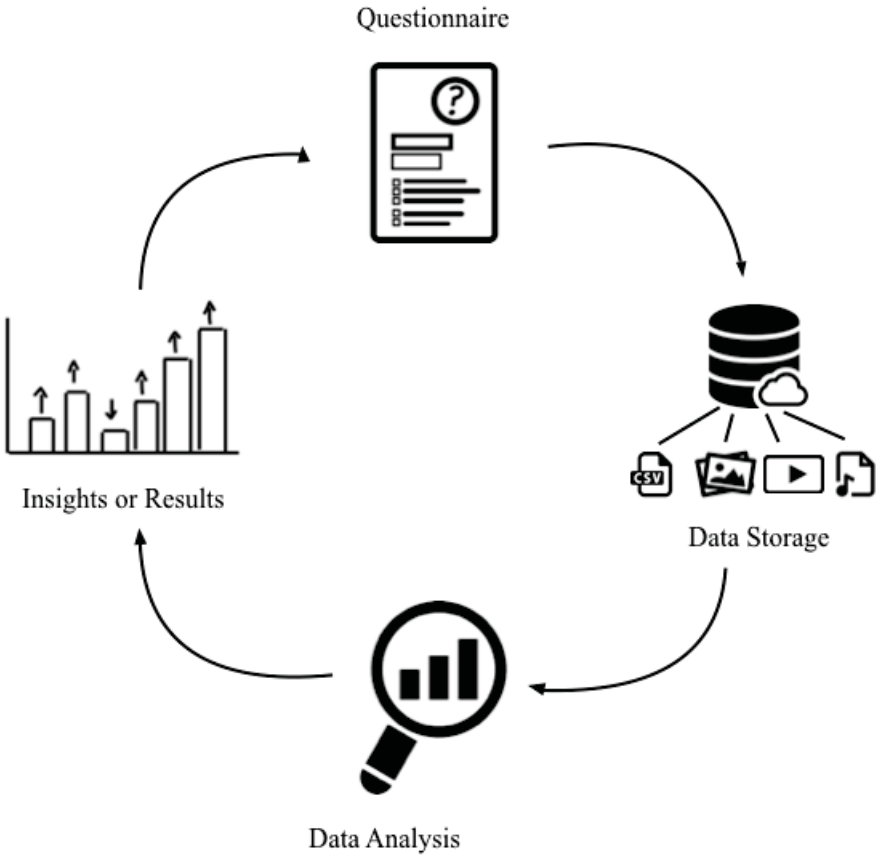

Figure 1.1: Stages of survey data anaysis

understand the whole context of large data. In such cases, summarization techniques can be used to summarize and give all the important information which we need to know. In the real world, these can be used to analyze open-ended survey questions, web crawling, processing emails automatically, etc.

1.4 Topic Modelling

Topic modelling is one of the frequently used text mining techniques which helps in the automatic coding of text data, corpus. It is a statistical model that can be used to identify the topics present in any given set of documents. It can also connect words which have similar meaning and distinguish between the various meanings of a word depending on their context. The corpus is categorized into themes which become the

topics of the corpus [10]. There is less human action involved which makes it easy for the researchers. The researcher has to input the number of topics to the algorithm which then gives the topic probabilities of the words and the topic distribution of the corpus. A topic is a collection of words which have semantic relatedness. This gives an idea of the distinct topics which are present in the collected data. For example, in a product feedback survey, people would have reported on the various features of the product. A topic modelling algorithm tries to find the co-occurrence of such patterns irrespective to the complexity of the sentence. A model considers the documents of a corpus as a bag-of-words (BOW) [11] from which the recurring co-occurrence patterns and topic distributions are found. By using topic modelling, we can classify the comments based on the features and analyze them to understand the customer concerns.

Traditional topic modelling algorithms consider the documents of a corpus as BOW. This makes the model ignore the order in which the words occur in the corpus. The model outputs the words in each topic which makes it difficult for the user to name the topics when they have minimum knowledge of the corpus domain. On the other hand, topic modelling is useful in automatic coding of a large corpus with minimum effort. It also paves way for understanding the corpus from a different perspective. Topic modelling can also be used to take a closer look at the data when applied to a small corpus. Finally, it helps in analyzing the text quicker, efficient and more objective.

1.5 Goals

To pave a healthier way forward for our future generations, it is important to understand the key factors involved in healthy aging. As the CLSA dataset contains potential ways to age healthier, it needs future processing to identify the various themes present. The dataset is substantial enough to make manual processing unfeasible. This requires some form of topic modelling techniques to extract the pattern. The two main goals for this thesis are outlined below.

- Find the various themes present in the dataset which can elucidate lay perspectives on what healthy aging means.

- Propose a topic modelling algorithm which will be able to seamlessly handle short, informal unstructured text data and generate meaningful topics or themes.

1.6 Thesis outline

A brief explanation of the thesis is outlined as follows. Chapter 2 provides a review of the methods used for analyzing qualitative survey data, traditional topic modelling algorithms and their issues while handling survey data. Chapter 3 gives an overview of the existing system and their drawbacks. It also gives a detailed description of our proposed system and its architecture. Chapter 4 gives an overview on CLSA and their dataset. It also explains the data pre-processing step. Chapter 5 discusses the experimental setup used for the comparison of the performance of our proposed model with the existing models. The results of each model are studied in depth to understand which model yields the best outcome. Finally, Chapter 6 concludes the thesis and exposes potential future work for this thesis.

Chapter 2

Background and Related work

This chapter discusses the background and related work in the field of qualitative research and topic modelling. It examines the existing models in the field of topic modelling and criticizes why these models do not work for short and unstructured text data.

Qualitative data is one form of data collected through open-ended survey questions. Such surveys are conducted on a focused group to take a closer look at the participant's perspective. The data collected through these surveys are usually unstructured text, images, videos or speech files. This makes it a challenge for the researchers to process and get useful insights. Different approaches can be used to retrieve information from such data. Rule based methods can be used to categorize the different themes present in the corpus, but it is a tedious task and is not portable to any other dataset. Labelling the dataset is time consuming, which makes is not suitable for predictive ML models. Since the dataset is small in size, there are not any suitable Deep Learning model for topic modelling.

2.1 The Grounded Theory

"The Grounded Theory" is one of the most popular methods to analyze qualitative survey data which was proposed by Barney et al. [12]. This approach helps to bring the theory and research goal closer by suggesting a logic to it. It provides four main stages which will help the researcher to test their hypothesis. The stages are code, concept, category and theory. In the coding stage, the researcher considers each response or data collected and assigns a key phrase or topic to it. In the concept stage, similar coded topics are grouped as one concept. For the category stage, the concepts are compared with each other to identify the links between them. If any connection is identified then they are grouped to form a category. The categories are compared to the data to learn when or why the events occured and to identify the

important properties of a category. If a concept is not supported by the data then it is dropped. Now, the collected data is divided into broader categories. In the final stage, the researcher tries to write a theory which will either approve or disapprove of their hypothesis.

2.2 Relevant Text Mining Background

2.2.1 Information Retrieval

The methodology proposed by the grounded theory is a manual form of the modern topic modelling techniques which can be classified as Information Retrieval. The concept of Information Retrieval [13] came into existence in the 1940's but it became a popular research area in the 1960's when a significant amount of work was done. Various methodologies were proposed, which can be used for document classification, document searches and exploring survey data.

Automatic Document Classification

In 1963, an algorithm to classify documents was proposed by Harold et al. [14]. They suggest, selecting keywords which are relevant to the collected documents and creating a correlation matrix, that contains the frequency of occurrence for each word in each document. Factor analysis is applied to the correlation matrix to reduce the dimensionality of the matrix to interpret patterns or useful information. To predict the category of the documents, the product of the normalized factor load and the number of occurrences of the keyword in that document is used. The category which has the highest value is been assigned to that document.

Experts labelled the documents manually to compare the performance of the documents classified. This method seems to be simple at computation but has a lot of manual activities which affects the accuracy of the model. The researcher should have a good knowledge of the documents to select the keywords which are a principal component of the algorithm. The next drawback is factor analysis because the dimensions used for factor analysis should be selected carefully as there are high chances that the most important features can be neglected during the process. Also, the label of the documents are purely based on individual knowledge which can also affect the

performance of the algorithm.

2.2.2 Topic Modelling

Topic Modelling is one of the most popular text mining techniques widely used in exploring unstructured text corpus. Text mining techniques became popular as technology took over the research surveys which generated large amount of data. Topic models enabled the researchers to inspect the corpus with less intervention. It saved effort and time spent on coding stage (First stage of the four stages in The Grounded Theory). This is an unsupervised method which helps in converting the corpus into meaningful topics from which information can be retrieved. The model gets an unstructured text data as input and calculates the probability of the words and topic distribution of the corpus. The algorithm outputs the top words for each topic to identify the topics hidden in the corpus.

Topic modelling became popular in 1990 after Deerwester et al.'s paper Indexing by Latent Semantic Analysis (LSA) [15]. This paper overcame the existing problems in the Information Retrieval and paved the way for topic modelling. LSA was not a probabilistic model. Hence, Thomas came up with his model named, Probabilistic Latent Semantic Analysis (PLSA) [16] which addressed the limitations of LSA. In 2003, a generative probabilistic model, Latent Dirichlet Allocation (LDA) [17] emerged as a popular topic modelling algorithm. Various approaches have been proposed to identify the topics in a corpus. Each has its own advantages and disadvantages which are discussed below.

2.3 Indexing by Latent Semantic Analysis

In the previous information retrieval methods, the keywords were chosen manually which affected the performance of the model. To improve the performance, LSA [15] suggested increasing the number of keywords selected. So the algorithm considered words which occurred in more than one document as a keyword. The drawbacks of predecessor models are synonymy, polysemy and that the words in a document are considered to be independent of each other. Synonymy means a word can be expressed in different ways. The words "positive" and "optimistic" both have similar meaning. If a word has more than one meaning, then it is known as polysemy. The

word “crane” means a bird as well as a construction equipment. These factors affect the recall and precision of the model.

To overcome these shortcomings, they used semantic similarities between the documents and terms rather than using hierarchical classification or factor analysis. Hierarchical methods failed to capture the semantics between documents and terms, whereas factor analysis methods are computationally costly. So they suggest a two-mode factor analysis model that uses Singular Value Decomposition (SVD) [18]. Using SVD, the documents and terms are converted into vectors and are placed in a desired dimensional space. The similarity between them can be found by calculating cosine or dot product. To test the model, it was applied to CISI and MED [19] datasets. Results showed that the model had better recall values when compared to the state-of-the-art models, SMART [20] and Voorhees systems [21]. It was also seen that the model had a low precision value which indicates that it performed poorly on polysemy.

2.3.1 Applications of LSA

Though LSA was proposed for document retrieval, it has applications in the area of classification, clustering, summarization etc. In 1996, a set of experiments were perfromed on text data which contained various sources of the Panama Canal construction by Peter [22]. In the first experiment, he made the participants to read the sources about Panama canal and asked them to summarize what they have read. SVD was applied to the source and summaries to generate the semantic matrix which was then converted to 100-dimensional space. By using cosine, the sources of the sentences in the summaries were found. To evaluate the results, he used two domain experts, who assisted in identifying the source of each of the line in the summary. The experts were allowed to assign more than one source for a summary sentence because there are chances that a sentence in the summary can be influenced by more than one document in the source as the documents are very closely related. If the sources assigned by the two experts matched then those sources become the source for that sentence in the summary. Results showed that the agreements between each expert and LSA were 56% and 49% respectively. This low agreement indicates that the documents have high semantic similarities.

In the second experiment, experts chose 10 sentences which they felt are the most important ones in the summary and their cosines were calculated. The experts also evaluated these sentences by assigning them cosine value "1.0", if the participant has reproduced the exact same line from the source and "0.0", if there is no similarity between the lines. The results showed that it performed well in the assignment grading as LSA was close to the expert's marking. This proves that LSA captures semantic similarities in the same way as humans.

The third experiment tried to calculate the coherence of the summaries written by the participants after reading articles about heart disease. LSA was used to calculate the coherence between each line of the summaries. The results were compared against a simple word overlap, which is similar to LSA. Unlike LSA, the dimension of the term-document matrix is unchanged in word overlap. Results showed that LSA was able to capture coherence from the basis of semantic similarity than just capturing words which were shared among sentences. These experiments show that LSA is good with unstructured text and works similar to a human expert in grading assignments, summarization and analyzing the coherence between the sentence using semantic similarity.

2.4 Probabilistic Latent Semantic Analysis

PLSA uses a statistical model known as aspect model [23]. This model is a latent variable model for co-occurrence of data. The model tries to associate each observation with an unobserved class variable $z \, \epsilon \, \{z_1, z_2, ..., z_k\}$. It uses Estimation Maximization (EM) algorithm for maximizing the likelihood estimation of latent variable, z. The model considers the documents, $D = \{d_1, d_2, ..., d_N\}$. The words present in the documents are converted to a set of unique words, vocabulary $W = \{w_1, w_2, ..., w_M\}$. PLSA does not consider the order in which the words occur in the document. The authors argue that considering the data as a BOW model still preserves most of the information. The co-occurrence table, $N \times M$ with $N = (n(d_i, w_j))_{ij}$ where $n(d, w) \epsilon N$ represents the frequency of word w in document d.

The model can be represented in two different ways based on the parameterization of the aspect model. The symmetric parameterization model which is shown in Fig. 2.1. In this model the latent variable or topic z acts as a connection between the

document d and word w. The second one is asymmetric parameterization as shown in Fig. 2.2. In this, the topic is selected first, to which a document and word are assigned. The authors argue that the models are statistically identical.

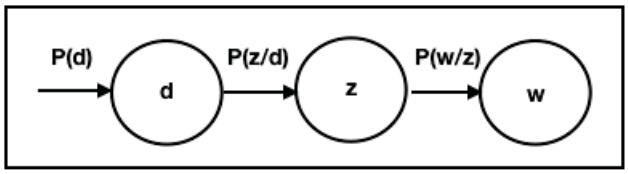

Figure 2.1: Graphical representation of Asymmetric parameterization for PLSA model

$$P(d,w) = P(d)\sum_{z\epsilon Z} P(w|z)P(z|d) \tag{2.1}$$

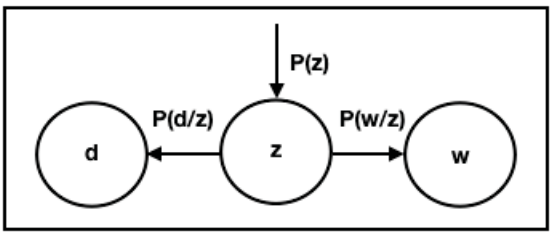

Figure 2.2: Graphical representation of Symmetric parameterization for PLSA model

$$P(d,w) = \sum_{z\epsilon Z} P(z)P(w|z)P(d|z) \tag{2.2}$$

The latent parameters can be estimated by using the maximum likelihood, Expectation Maximization (EM). EM has two main steps, expectation, E-step which calculates the probabilities of latent variables and a maximization, M-step which updates the latent parameters.

E-step:

$$P(z|d,w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z' \epsilon Z} P(z')P(d|z'P(w|z')} \tag{2.3}$$

M-step:

$$P(w|z) \propto \sum_{d \epsilon D} n(d,w)P(z|d,w), \tag{2.4}$$

$$P(d|z) \propto \sum_{w \epsilon W} n(d,w)P(z|d,w), \tag{2.5}$$

$$P(z) \propto \sum_{d \epsilon D} \sum_{w \epsilon W} n(d,w)P(z|d,w) \tag{2.6}$$

The probability of word belonging to a topic is given by Eq. 2.4 and that of a document belonging to a topic is given by Eq. 2.5. The topic probability $P(z)$ is represented in Eq. 2.6. The E and M steps are performed iteratively until it reaches a convergence or when the parameters are no more updated meaning there is no room for improvement.

When a corpus or set of documents are passed to the PLSA, it outputs the top words which belong to each topic by following the three steps.

- Selects a document, d_i with probability $P(d_i)$
- Selects a latent variable z_k with probability $P(z_k|d_i)$
- Generates a word w_j with probability $P(w_j|z_k)$

Since PLSA has a stronger statistical foundation, which makes it perform better than LSA by overcoming the polysemy issue.

2.5 Latent Dirichlet Allocation (LDA)

LDA is one of the most famous Topic modelling algorithm and is considered as one of the state-of-the-art models. PLSA's major drawbacks are that the number of parameters grows linearly as the size of the corpus increases which may cause overfitting. Both LSA and PLSA consider the documents as a BOW model, which do not consider the order of words in the document and the order of the documents in the corpus. According to de Finetti [24], every collection of an exchangeable random variable can

be represented as a mixture distribution, which means the model should be able to consider both the documents and words exchangeable. So LDA tries to capture the intra-document statistical structure using mixture distribution.

LDA is a three-level hierarchical Bayesian model. A Bayesian model predicts the probability of an event based on prior knowledge. The documents are modelled to give a finite mixture of topics which are in turn carved to give an infinite mixture of topic probabilities. LDA handles any given input as words, documents and corpus. A word is a basic unit of the data and is a part of the vocabulary $V = \{1, ..., V\}$. A document is a collection of N words, $\mathbf{w} = \{w_1, w_2, ...w_N\}$. Finally the corpus is a collection of M documents $D = \{\mathbf{w_1}, \mathbf{w_2}, ...\mathbf{w_M}\}$. LDA also has two Dirichlet priors α and β which represent the per document topic distribution and per topic word distribution respectively. θ represents the topic distribution per document. The dimensionality k is assumed to be fixed and known. It has three main assumptions about every document present in the corpus.

1. Choose N $\sim$ Poisson(ξ).

2. Choose $\theta \sim$ Dir(α).

3. For every N words w_n, choose a topic (z_n) $\sim$ Multinominal (θ) and choose a word w_n from $p(w_n|z_n, \beta)$.

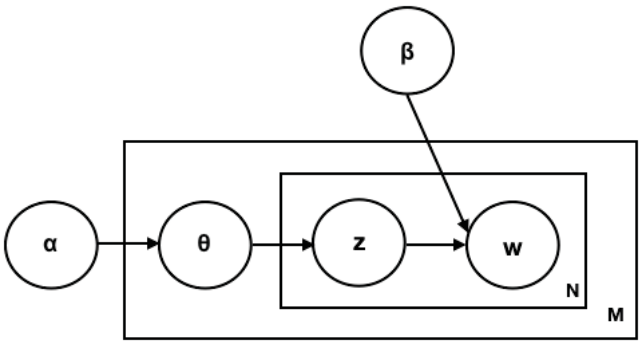

Figure 2.3: Graphical model of LDA

From Fig. 2.3, it is seen that α and β are corpus-level variables which are sampled once during corpus generation. θ is a document-level variable which is sampled once for every document and z and w are word-level variables which are sampled for every word in the document. Given the values of α and β the joint distribution of the model is obtained by

$$P(\theta, z, w|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} P(z_n|\theta) P(w_n|z_n, \beta) \tag{2.7}$$

The marginal distribution of the document is given by

$$p(\mathbf{w}|\alpha, \beta) = \int p(\theta|\alpha) (\prod_{n=1}^{N} \sum_{z_{dn}} p(z_n|\theta) p(w_n|z_n, \beta)) d\theta \tag{2.8}$$

To get the probability of the corpus, the individual probability of the documents are multiplied.

$$p(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) (\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_n|\theta) p(w_n|z_n, \beta)) d\theta_d \tag{2.9}$$

2.6 Topic Modelling for Short Text

LDA considers documents as a mixture of topics and topics as a probability distribution over the documents. The topics are discovered by identifying the document-level co-occurrence of the words. When these methods are applied to short text, the model suffers from sparsity problem. This is because there is not much data available when the length of the document is short. Various methods have been proposed to solve the data sparsity problem by extending text in the dataset. TwitterRank [25] suggested to increase the length of the document by collecting tweets from one user account and consider that as a single document. When there are not enough tweets from the same person, tweets of which belong to the same domain can be aggregated to form a document [26]. Aggregation of data are also not always possible because information is not available for certain domains.

2.6.1 Biterm

Biterm(BTM) model [27] was proposed to overcome the problems of LDA and all the other models which have been proposed to address the topic modelling problems for short text. The authors argue that the word co-occurrence patterns should be studied at corpus-level rather than document-level which will solve the data sparsity problem suffered by the traditional topic modelling algorithms. The documents are converted to biterms. Two words in a document are combined to form a biterm. Eg. Maintain

a good diet, biterms in this sentence are “maintain a”, “a good”, “good diet”, “diet maintain”. These biterms are sent to the model rather than tokens (single word). This aids in identifying the word co-occurrence in the corpus. Unlike LDA, BTM considers the whole corpus as a mixture of topics and each biterm is assigned to a topic independently. BTM follows a three-step generative process.

1. For every topic z, a topic-specific word distribution is drawn, $\phi \sim Dir(\beta)$

2. For the corpus, a topic distribution is drawn, $\theta \sim Dir(\alpha)$

3. For every biterm b present in biterm set B, a topic assignment, $z \sim Multi(\theta)$ and two words, $w_i, wj \sim Multi(\phi_z)$ are drawn.

The joint probability of a biterm is given as

$$P(b) = \sum_z P(z)P(w_i|z)P(w_j|z) = \sum_z \theta_z \phi_{i|z}\phi_{j|z} \tag{2.10}$$

and for the biterm set as

$$P(B) = \prod_{i,j} \sum_z \theta_z \phi_{i|z}\phi_{j|z} \tag{2.11}$$

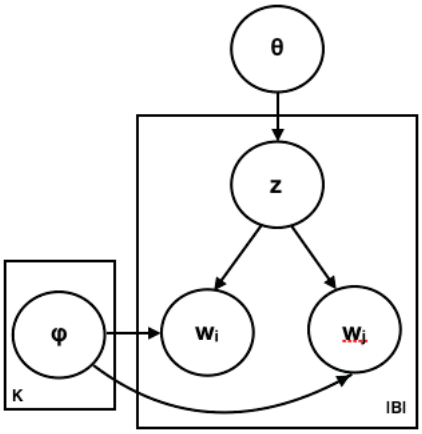

Figure 2.4: Graphical representation of BTM model

By using the three-step generative process, BTM overcomes the sparsity problem and is also able to identify multiple topics in the document and the correlation between the words in the documents. The topic of a document is given by

$$P(z|d) = \sum_b P(z|b)P(b|d) \tag{2.12}$$

$$P(z|b) = \frac{P(z)P(w_i|z)P(w_j|z)}{\sum_z P(z)P(w_i|z)P(w_j|z)} \tag{2.13}$$

$$P(b|d) = \frac{n_d(b)}{\sum_b n_b(b)} \tag{2.14}$$

where $P(z) = \theta_z$, $P(w_i|z) = \phi_{i|z}$, $n_d(b)$ is the number of times biterm b has occurred in the document d. Usually $P(b|d)$ is a uniform distribution in short texts.

BTM uses Gibbs Sampling [28] to estimate the 3 latent variables z, θ and ϕ. By using collapsed Gibbs sampling, θ and ϕ are integrated out because of the latent priors α and β. Only z has to be sampled for a biterm b given the other variables and is given by

$$P(z|z_{-b}, B, \alpha, \beta) \propto (n_z + \alpha)\frac{(n_{w_{i|z}} + \beta)(n_{w_{j|z}} + \beta)}{(\sum_w n_{w|z} + M\beta)^2} \tag{2.15}$$

where n_z is the number of times the biterm has been assigned to z and $n_{w|z}$ is the number of times the word w is assigned to z. The latent variables ϕ, topic-word distribution and θ, global topic distribution is given by

$$\phi_{w|z} = \frac{n_{w|z} + \beta}{\sum_w n_{w|z} + M\beta} \tag{2.16}$$

$$\theta_z = \frac{n_z + \alpha}{|B| + K\alpha} \tag{2.17}$$

where total number of biterms is given by $|B|$.

BTM captures the word co-occurrence patterns by using biterms. These biterms are considered as BOW model which means the order of occurrence of biterms in the document is not considered and hence cannot understand the context of the words. Also, the relationship between the words of a biterm is not considered. A biterm containing words which do not have any relationship between them and occurs frequently, there are chances that the algorithm can give significant importance to it.

Chapter 3

Methodology

This chapter introduces the techniques used for topic modelling model which will help us understand the various themes or recurring patterns present in the data obtained from CLSA. The rest of the chapter is designed to discuss the implementation of the proposed model, N-gram based keyword topic model.

3.1 N-grams

N-grams are an adjacent sequence of items in any given sentence. Unigram contains one item, bigram has two items, trigram consists of three items and so on. The items can be words, bytes, characters or syllables. N-grams are commonly used in predictive analysis [29] and identifying context because of its sequential nature. When n, the number of items, is set to a large value, n-grams may occur only once as they may contain the whole word or sentence. This may result in a large n-gram set which may affect the performance of the model.

3.1.1 N-gram derivation

As mentioned before, n-gram items can be characters or words. When the items are characters it is known as character n-gram. In order to convert a document into n-grams, one should use a sliding window concept as shown in Fig. 3.1. Character n-grams consider spaces between the words as a character as well. It helps to be robust to typographical errors.

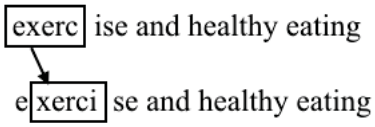

Figure 3.1: Character N-grams extraction

If the items are words, then it is known as word n-gram. In this, only the words are included, unlike character n-gram. When n is greater than 1, more than one word will be included in an n-gram which will help in context detection. A 3-word n-gram extraction is shown in Fig. 3.2

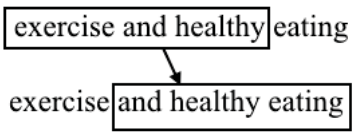

Figure 3.2: Word N-gram extraction

3.1.2 Applications of N-grams

N-grams play a vital role in Statistical Natural Language Processing. It helps in spelling correction, document clustering, language detection, authorship attribution, understanding context, automatic grading and many other tasks. Some of the applications of n-grams are discussed briefly below.

Spelling Correction

We, humans, are prone to make typographical errors while typing. This may be because of pressing the wrong key. Mostly these errors are minor and can be predicted easily by using character n-grams. When a sentence is converted into character n-grams, we have n-grams which capture the sequence of the letters which form words. Using this we can identify the errors in the words. Typographical errors can be corrected by a slight modification to the misspelt words like insertion, deletion, substitution and reversal [30].

Document Clustering

Documents belonging to the same domain can be clustered by using n-grams [31]. Each domain has its own set of technical words which makes it unique from the others. By taking advantage of this, n-grams will be able to identify the documents which belong to the same domain. For example, let us consider a set of documents which talk about cricket and football. Though both are sports, cricket has some

specialized words like spin, leg-before-wicket, run-out which helps in differentiating it from football which has its own technical words like touchdown, punt, free kicks.

Language Detection

Every human language has a structure and n-grams are good at capturing these structures [32]. Each language has a different set of top n-grams. Eg. words like 'the', 'be', 'are', 'and' occur frequently in the English language. In any document, we can find the most frequently occurring n-gram which will help in identifying the language of that document.

Authorship Attribution

N-grams can also be used to identify the authors of articles, papers or any type of literature [33]. Every person has a different style of writing and tends to use certain words predominantly than the others. By using character n-grams, we can create an n-gram profile for each author based on their top n-grams. The top n-grams of documents can be compared to existing n-gram profiles to identify the author. This becomes useful when some ancient literature does not have the author information. This may not work if there is a change in the style of writing.

3.2 Token Keyword Model

Token Keyword Model (TKM) [34] proposed by Schneider et al., is a topic modelling algorithm which tries to overcome the drawbacks of the existing models like PLSA, LDA and LSA. This model considers the frequency of a word within a topic and also among the topics, unlike the predecessors. This is because if a word is present in all the topics then it might have a higher probability for every topic. The model also takes into account the context of the documents whereas the previous ones converted the documents into a BOW which resulted in a random assignment of words to a topic. TKM model uses a novel algorithm to overcome all these issues.

The model calculates a keyword score for each word present in that document. This is done by taking into account the commonality of the word within a topic and as well as among all the other topics. Since this model accounts for the order of

the words in a document, the keyword score assigned to each occurrence of a word depends on its neighbouring words. This means, if a word occurs more than once in a document with different neighbouring words then for each occurrence, a different keywords score is assigned. The topic document distribution is the sum of all the keyword score of the words present in that document. This is similar to the existing models. Finally, the model tries to identify the number of topics present in the corpus. This becomes useful when there is minimal knowledge about the domain.

Model

The core idea of TKM is from the aspect model [35], which determines the joint probability of $D \times W$, where D is the corpus whose documents are represented by d and W is the set of unique words (dictionary) and the words are denoted by w. The total number of words in a document is given by $|d|$. The model considers that the words and documents are conditionally independent given a topic.

$$p(d, w) := p(d).p(w|d)$$

$$p(w|d) := \sum_{t} p(w|t).p(t|d) \tag{3.1}$$

TKM model uses this aspect model but includes the position of the word in the document when calculating the keyword score $f(w, t)$. This helps in understanding the context of the document and is indicated by i, the position of the word in the document.

$$p(d, w, i) := p(d).p(i|d).p(w_i = w|d, i) \tag{3.2}$$

$$p(w|d, i) := max_{t,j \epsilon R_i}(f(w, t) + f(w_{i+j}, t)).p(t|d) \tag{3.3}$$

$$\text{where, } R_i := [max(0, i - L), min(|d| - 1, i + L)]$$

$$p(t|d) := \frac{(\sum_{i \epsilon [0,|d|-1]} f(w_i, t))^{\alpha}}{\sum_t (\sum_{i \epsilon [0,|d|-1]} f(w_i, t))^{\alpha}} \tag{3.4}$$

The probability distribution, $p(d)$ is proportional to $|d|$ and $p(i|d)$ is assumed to be a uniform distribution as there is no explicit importance given to any particular word position in the document. The probability of a word in a particular position is calculated using Eq. 3.3. TKM model uses the keyword score, $f(w,t)$ rather than $p(w|t)$ which depends on the frequency of the word in a topic. The keyword score is assigned to a word only if it has a significant impact on the topic and occurs frequently. The score depends on $p(w|t), p(t|w)$ and $p(t)$ which tries to impose the topic of the word with high frequency to its neighbouring L words. The topic assignment of a word w_i depends on all the w_{i+j} where $j \in [-L, L]$. For start and end of a document d, the boundary $j \in [max(0, i-L]), min(|d|-1, i+L)]$. When a word is weakly associated with a topic, its score is close to zero even if its nearby words have a high keyword score for that topic. The algorithm also assumes that each occurrence of a word comes from only one topic which is done by taking the maximum of Eq. 3.3.

To compute the topic of a document $p(t|d)$, the aggregate of keyword score $f(w,t)$ of the words of the document is taken. The parameter α from Eq. 3.4 determines the number of topics present in the document. If the value of α is large, It means that there are fewers topics present in the document.

Keyword score calculation

To calculate keyword score $f(w,t)$, two aspects are considered namely frequency of the word $n(w,t)$ and importance of the word in that topic $p(t|w)$. A uniform distribution of $p(t|w)$ states that the word is not significant to any topic. To find the importance of a word w the inverse of entropy $H(w)$ is calculated.

$$H(w) := -\sum_{t} p(t|w).\log(p(t|w)) \tag{3.5}$$

To get a uniform distribution the entropy is maximized i.e., $p(w|t) = 1/|T|$ which will give $H(w) = \log|T|$ where $|T|$ - number of topics and $1/H(w)$ determines the topic to which a word belongs. When a word is always assigned to the same topic, its entropy is zero which results in infinity when inverted. To avoid this, 1 is added to denominator $1/1 + H(w)$. If a word occurs fewer than number of topics $|T|$, then its entropy is $\log\ n(w) < \log|T|$. According to the authors, this cannot be ignored as it might result in a high keyword score for a rare word whose each occurrence may be

assigned to different topics. This can be avoided by using factor log min($|T|, n(w)+1$). One is added to $n(w)$, to ensure that no word has non-zero weights. The concentration score is given by

$$con(w) := \left(\frac{\log(min(|t|, n(w)+1)}{1+H(w)}\right)^{\delta} \tag{3.6}$$

The keyword score also includes the frequency of the word in a topic which can be calculated by taking the probability of the word times the total number of documents; i.e., $p(w,t) \cdot \sum_{d \in D} |d|$. Since this is a classification task, damped frequencies work better. This is because classification depends on concentration to understand if a word belongs to a topic or not. Calculating keyword score $f(w,t)$ this way may produce words which can be understood by domain experts only. So $f_{hu}(w,t)$ is used which predominantly considers the frequency of the words.

$$f(w,t) \propto \log(1 + (p(w,t) \cdot \sum_{d \in D} |d|) \cdot con(w) \tag{3.7}$$

$$f_{hu}(w,t) \propto (p(w,t) \cdot \sum_{d \in D} |d|) \cdot con(w) \tag{3.8}$$

Inference

The parameters which will maximize the data likelihood can be found by using an inference algorithm. Gibbs sampling cannot be used because of the complexity of the algorithm. Instead, Expectation-Maximization (EM) [36] along with standard probabilistic reasoning on the word-topic assignment frequencies. In the E-step, the latent variable is estimated i.e., $p(t|w,i,d)$ and in the M-step the loss function is maximized with respect to the parameters used in $p(t|w,i,d)$. By modifying the equation 3.3.

$$t(w,i,d) := argmax_t\{(f(w,t) + f(w_{i+j},t)).p(t|d)|j \epsilon R_i\}$$

where, $t(w,i,d)$ denotes the topic of a word in a particular context.

$$p(t|w,i,d) = \begin{cases} 1 & t_{w,i,d} = t \\ 0 & t_{w,i,d} \neq t \end{cases}$$

(3.9)

According to [37], when there is only one latent variable for every observation i.e., every document is assigned to one topic {D,t} which makes it a complete dataset. Generally we don't get the complete dataset, we only have the documents. The value of t, topic is considered to be given by posterior distribution $p(t|D,\theta)$. Since we do not have the complete data log-likelihood, we use E-step of EM algorithm to calculate the expected value under the posterior distribution of the latent variable. Then the value is maximized in the M-step. The current value of the latent variable is denoted by θ^{old} and the value obtained after the EM algorithm as θ^{new}. The latent variable is arbitrarily assigned to a starting value.

E-Step:

$$Q(\theta,\theta^{old}) = \sum_{d,i,t} p(t|D,\theta^{old})\ (D,t|\theta) \tag{3.10}$$

M-Step:

$$\theta^{new} = argmax_{\theta} Q(\theta,\theta^{old}) \tag{3.11}$$

To calculate the value of $p(w|t)$, $p(t)$ and $p(w)$, an assumption is made that each word w is given a topic t for a given set of documents D is $n(w,t)$. Using empirical distribution, the probability of a word given a topic is

$$p(w|t) := \frac{n(w,t)}{\sum_w n(n,t)} \tag{3.12}$$

where, $n(w,t)$ is calculated by adding up the values from E-step i.e., $p(t|w,i,d)$ because of the assumption every occurrence of the word is assigned to only one topic.

$$n(w,t) := \sum_{t,d}\ p(t|w,i,d) \tag{3.13}$$

$p(t|w)$ is calculated using Bayes' law,

$$p(t|w) = \frac{p(w|t)\cdot(t)}{p(w)} \tag{3.14}$$

where, $p(t) = \frac{\sum_w n(w,t)}{\sum_{w,t} n(w,t)}$ and $p(w) = \frac{\sum_t n(w,t)}{\sum_{w,t} n(w,t)}$

$$p(t|w) = p(w|t)\cdot\frac{p(t)}{p(w)}\ = \frac{n(w,t)}{\sum_t\ n(w,t)}$$

Finally to calculate the keyword score $f(w,t)$, using equation 3.7 where $\sum_{d \epsilon D} |d| = \sum_{w,t} n(w,t)$ as a word belongs to only one topic.

$$f(w,t) \propto \log(1 + \ (p(w,t) \ . \ \sum_{d \epsilon D} |d|)(w)$$

$$\propto \log(1 + p(w|t) \cdot (t) \cdot \sum_{w,t} n(w,t) + \beta)(w)$$

By substituting the values for $p(w|t)$ and $p(t)$ and simplifying it we get,

$$\propto \log(1 + n(w,t) + \beta)(w)$$

3.3 TKM System Architecture

Fig. 3.3 shows the system architecture of the TKM model. The raw data is converted into tokens and is pre-processed by stemming and removing stopwords and words which occur only once in the corpus. The clean data is now sent to the TKM model. The model calculates the keyword score of the words present in the documents. Then the topic of a document is calculated by aggregating the keyword scores of the words present in that document. In TKM, the number of topics k is determined by the model. It then generates the top words which belong to each topic.

TKM overcomes the issues of the existing systems, as it does not consider the documents as a BOW model, unlike its predecessor models. Hence it is good at capturing the context of the documents. Another advantage of the model is that it takes into account the words which occur commonly in all the topics. This means that it is able to identify the inter and intra frequencies of words in the topics. But when it comes to a short, messy and small sized dataset, TKM fails to group documents which are understandable by humans. TKM also does not work well with informal sentences which contain typographical errors.

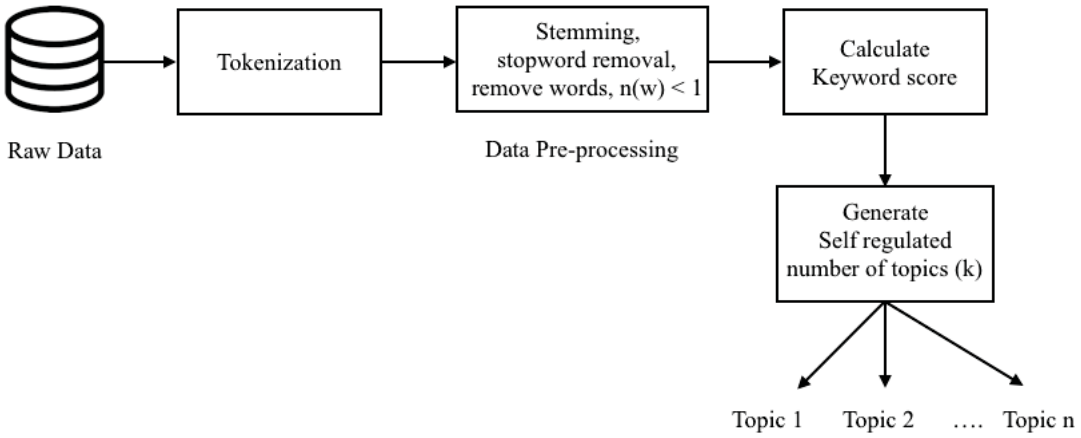

Figure 3.3: Token Keyword Model system architecture

3.4 Proposed System Architecture

To overcome the issues of TKM, we propose a system with modifications to the existing TKM model. The proposed model is designed to handle typographical errors, grammatically incorrect sentences and short sized datasets. Fig. 3.4 shows the overview of the system we propose to tackle the problems of TKM. The model consists of four steps which include data pre-processing, extracting N-grams, calculating keyword score and finally generating k number of topics.

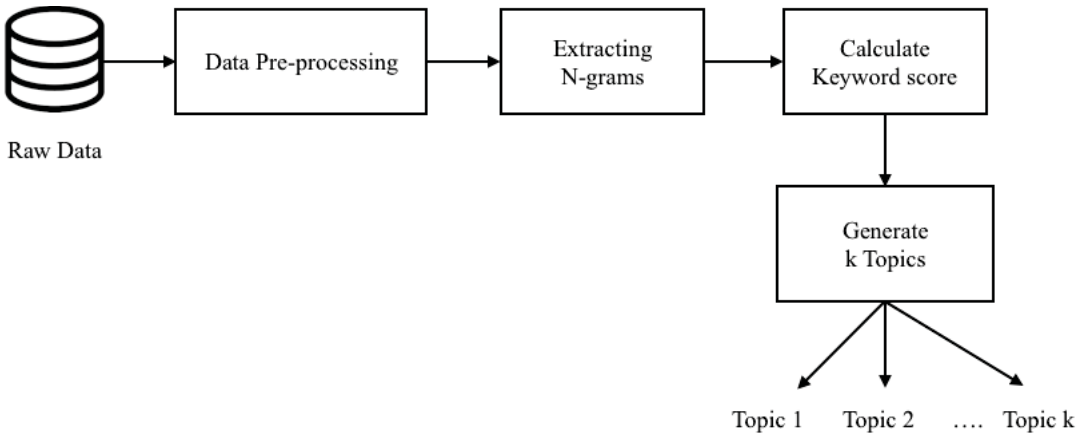

Figure 3.4: Proposed Topic Modelling System

In the data pre-processing phase, punctuation and non-UTF-8 compatible characters are removed. Pre-processing of the data is explained in detail in Chapter 4. Character n-grams are then extracted from the pre-processed data. N-grams which

occur less than five times are removed and the remaining n-grams form the n-gram vocabulary. Then keyword score is calculated, which determines the topic of an n-gram as shown in Algorithm 1. Unlike TKM, the proposed model does not generate its own number of topics for the corpus. The number of topics, k is given as an input to the model and it generates topics along with the top n-grams which belong to that topic.

Let's consider a document d, "happiness and moderation". When extracting n-grams with $n = 6$, the n-gram set

$$NG \subseteq \{\text{"happin"}, \text{"appine"}, \text{"ppines"}, \text{"piness"}, \text{"iness "}, \text{"ness a"}, \text{"ess an"}, \text{"ss and"}, \text{"s and "}, \text{" and m"}, \text{"and mo"}, \text{"nd mod"}, \text{"d mode"}, \text{" moder"}, \text{"modera"}, \text{"oderat"}, \text{"derati"}, \text{"eratio"}, \text{"ration"}\}$$

According to our proposed system, the keyword score of n-gram ng_i depends on the L n-grams present to left and right of it. If $ng_i =$ "ss an", then it's keyword score depends on ng_j, seven n-grams present to the right ("s and ", " and m", "and mo", "nd mod", "d mode", " moder", "modera") and left ("happin", "appine", "ppines", "piness", "iness ", "ness a", "ess an"). But if ng_i is does not have enough number of n-grams before and after it then we consider the edge rule where $j \in R_i := [max(0, i - L), min(|d| - 1, i + L)]$. For example, if $ng_i =$ "happin", since it does not have any n-grams present to the left of it, the algorithm will only consider the once which are present to the right ("appine", "ppines", "piness", "iness ", "ness a", "ess an", "ss and") of it. To calculate the topic of the document d, the aggregate of keyword score of all the n-grams present in it are considered.

Table 3.1 shows the notations used in algorithm 1. The algorithm shows how the proposed system works. The documents are converted to n-grams and sent to our proposed algorithm which will classify them into topics. The number of neighbouring n-grams which will affect the topic of an n-gram, ng_i is denoted as L and is initialized to 7. The probability of the topic given a particular document, $p(t|d)$ is a fraction of the number of given topics, k i.e., $p(t|d) = 1/k$. The probability of an n-gram for a topic, $p(ng|t)$ is a fraction of the number of n-grams in the vocabulary i.e., $p(ng|t) = 1/|NG|$. When an n-gram is not assigned to any topic, the frequency of the n-gram for that topic is initialized to zero, $n(ng, t) = 0$. For every n-gram, ng_i in

Algorithm 1 Algorithm of the proposed system

$L := 7$; $p(t|d) := 1/k$; $T := [1, k]$; $p(ng|t) := 1/|NG| + noise$

while $p(ng, t)$ "not converged" **do**

$n(ng, t) := 0$

for $d \; \epsilon \; D$ **do**

for $i = 0 \; to \; (|d| - 1)$ **do**

$\{R_i$ denotes the edges of the document$\}R_i := [max(0, i - L), min(|d| - 1, i + L)]$ $t(ng, i, d) := argmax_t\{(f(ng_i, t) + f(ng_j, t)) \cdot (t|d)|j \; \epsilon \; R_i\}$

$n(ng_i, t(ng_i, i, d)) = n(ng_i, t(ng_i, i, d)) + 1$

end for

end for

$p(t|ng) := \frac{n(ng,t)}{\sum_{t'} \; n(ng,t)'}$

$H(ng) := -\sum_t p(t|ng).\log(p(t|ng))$

$con(ng) := \left(\frac{\log(min(|T|, n(ng)+1)}{1+H(ng)} \right)^{\delta}$

$f_{hu}(ng, t) := \frac{n(ng,t)(ng)}{\sum_{ng'} n(ng',t)(ng')}$

$p(t|d) := \frac{(\sum_{i \epsilon [0,|d|-1]} f(ng_i,t))^{\alpha}}{\sum_t (\sum_{i \epsilon [0,|d|-1]} f(ng_i,t))^{\alpha}}$

end while=0

Symbol	Meaning
D	Corpus
d	document from D
$\|d\|$	number of n-grams in document d
ng	n-gram
ng_i	i^{th} n-gram in the document d
k	number of topics
T	set of topics, T $\subseteq$ [0, k-1]
t	topic t in T
$t(ng, i, d)$	topic of the ith n-gram in document d
α, β	topic, n-gram prior
δ	weight for n-gram concentration
$n(ng, t)$	number of assignments of n-gram ng to topic t
$n(ng)$	number of occurrences of n-gram ng in D
L	number of neighbouring n-grams to be considered

Table 3.1: Notations

the document d in the corpus D, we assume that it belongs to only one topic. Unlike traditional generative probabilities, the keyword score accounts for the frequency of the n-gram as well as the importance of the n-gram in the given topic. Hence the topic of an n-gram is calculated by adding the keyword score value of the L neighbouring n-grams and maximizing it.

$$t(ng, i, d) := argmax_t\{(f(ng, t) + f(ng_{i+j}, t))\cdot (t|d)|j \ \epsilon \ R_i\} \tag{3.15}$$

where, $R_i := [max(0, i - L), min(|d| - 1, i + L)]$ which denoted the edges of the n-grams collection for the document.

The frequency of the n-gram ng_i belonging to a topic is updated by adding one. This helps in understanding how often an n-gram is assigned to a topic. If every occurrence of the n-gram is assigned to the same topic then it means that the n-gram is characteristic to that topic. The probability of topic for a given document is calculated by aggregating the keyword score of all the n-grams present in that document.

$$p(t|d) := \frac{(\sum_{i\epsilon[0,|d|-1]} f(ng_i, t))^\alpha}{\sum_t (\sum_{i\epsilon[0,|d|-1]} f(ng_i, t))^\alpha} \tag{3.16}$$

To calculate the keyword score, we consider the frequency of the n-grams in the topic $n(ng, t)$ and how important a word is to a topic, $p(t|ng)$. If $p(t|ng)$ has a uniform

distribution, it means that it is present equally in all the topics which does not make it significant to any topic. To find how characteristic an n-gram is to a topic the concentration is calculated. For this we find the inverse of entropy by using

$$H(ng) := -\sum_{t} p(t|ng).\log(p(t|ng)) \tag{3.17}$$

To attain uniform distribution, entropy is maximized i.e., $p(ng|t) = 1/|T|$ which gives $H(ng) = \log|T|$ where $|T|$ is the number of topics and $1/H(ng)$ determines the topic to which a n-gram belongs to. One is added to the denominator, $1/1 + H(ng)$ because when an n-gram is always assigned to the same topic the entropy becomes zero which will result in infinity when inverted. Though n-grams which occur less than 5 times in the corpus are removed, if the frequency of the n-gram is less than the number of topics $|T|$, its entropy can be at most $\log\ n(ng) < \log|T|$. This cannot be ignored as it might result in a high keyword score for a rare n-gram whose each occurrence may be assigned to different topics. This can be avoided by using factor $\log min(|T|, n(ng) + 1)$. One is added to ensure that no word has non-zero weights. The concentration score is given by

$$con(ng) := \left(\frac{\log(min(|t|, n(ng) + 1)}{1 + H(ng)}\right)^{\delta} \tag{3.18}$$

The keyword score depends on the frequency of the n-gram in the topic which is the probability of the n-gram times the total number of n-grams ($p(ng, t) \cdot \sum_{d\epsilon D}|d|$). To make the words represented by the n-grams understandable to everyone we use the raw frequency of the n-grams in a topic. The keyword score is calculated by using

$$f_{hu}(ng, t) \propto (p(ng, t)\ .\ \sum_{d\epsilon D}|d|)(ng) \tag{3.19}$$

Chapter 4

Dataset and Pre-processing

This chapter will give a detailed description of the dataset used in this thesis. The rest of the chapter discusses the pre-processing techniques used for cleaning the dataset.

4.1 What is CLSA?

The Canadian Longitudinal Study on Aging (CLSA) is one of the strategic initiatives of the Canadian Institute of Health Research (CIHR). Led by three principal investigators, it follows over 51,000 Canadian men and women aged 45-85 at entry, every three years, for a total of 20 years. The objective of the CLSA is to understand the interplay the wide range of factors that influence aging, and to study the trajectories of aging among Canadians [38].

CLSA is conducting a longitudinal study on healthy aging. Longitudinal study which means observing the same variables for a long time period and trying to find behaviour patterns. The main goal of CLSA is to understand the aging process among older Canadians. The data collected can be used for various researches which can help to answer research questions like how some people are aging well while others do not, predict diseases at an early stage, improve health services and policies for a better aging process in humankind in Canada and around the world.

The study also tries to consider how non-medical factors like economic status and social activities affect aging. For this purpose, the participants are going to be followed up every three years with the same set of questions or have a couple of questions added to the previous ones based on the needs of the research programme.

4.2 Data Type

The data collected from these interviews are unidentified to protect the identity of the participants and are made available to the researchers. The data collected using

entity_id	AGE_NMBR_TRM	SEX_ASK_TRM	SDC_COB_TRM	GEN_HLAG_TRM
17724724	46	F	001	exercise and proper diet laughter
88060186	60	F	001	trying your best to live in moderation social
49119706	61	F	001	mental and generally good health

Table 4.1: Sample data collect during the interview

Variable	Name	Missing	Label
SEX_ASK_TRM	M	0	Male
SEX_ASK_TRM	F	0	Female
SDC_COB_TRM	001	0	Canada
SDC_COB_TRM	002	0	United Kingdom
SDC_COB_TRM	003	0	United States of America
SDC_COB_TRM	777	1	Missing

Table 4.2: Sample data from Category table

telephonic and in-home interviews are stored in separate CSV files. For research purpose, the questions are converted to columns for analysis and the answers of each participant form a record as shown in table 4.1. Every participant has a unique identification number as the records are de-identified to protect the participant's privacy. Each column has a coded abbreviation whose information is stored in a separate file. The file which contains the details of the questions has two sheets, Variable and Categories sheets. The Variables sheet has details like data type, the label which gives information on the answer, the comment has details which may be useful for the researcher and question which was actually asked the participant as shown in table 4.3. The answers are converted into numerical data for the ease of use by the researchers. This is stored in the category sheet along with their explanation as in table 4.2.

Name	Value Type	Label	Comment	Question
AGE_NMBR_TRM	integer	Age (years)	Calculated: Date of interview less reported Date of Birth. The few cases of ages outside the study population range (45-85) are due to time lapse issues between the initial recruitment stage and the actual date the interview was completed.	What is your age?
SEX_ASK_TRM	text	Sex		Are you male or female?
SDC_COB_TRM	text	Country of birth	Includes additional categories based on open text responses of other countries of birth variable.	In what country were you born?
GEN_HLAG_TRM	text	Promote healthy aging verbatim		I have talked with many adults and learned something from each of them about what they think, promotes healthy aging. What do you think makes people live long and keep well?

Table 4.3: Sample data from Variable table

4.3 Characteristics

The CLSA interviewed approximately 50,000 Canadians between the ages 45-85 years for the purpose of this study. These participants will be followed until 2033 or their death. The two major modes of collecting data were through telephone and in-home interview. Overall 21,242 participants took a telephone interview which lasted roughly 60-90 minutes. For the in-home interview, there were around 30,098 participants and the interview lasted for roughly 90 minutes. Among them, 84.1% of the participants were born in Canada with 91.8% recognized as white. The average age of the participants was 62.9 out of which 50.9% of them were females and 69.7% of them were married/common-law and 90.8% of them had a post-secondary education.

As our research tries to use the perspectives of Older Canadians to analyze what they value for 'healthy aging', we use the verbatim response to the question *"I have talked with many adults and learned something from each of them about what they think promotes healthy aging. What do you think makes people live long and keep well?"*. The Older Canadians responded to this question by using one to 319 words with a mean of 12.7 and SD of 14.4 words. The data collected contains 51,340 records. Some participants refused to answer the questions and hence these responses were removed. We consider only english responses for the scope of this thesis. The responses from the users are phrases which are not fully structured sentences and grammatically correct. Since the data was entered manually, it has a lot of typographical errors.

4.4 Preprocessing

Preprocessing is the process of converting the raw data into an understandable format as the raw data is incomplete, inconsistent or contains human errors like spelling mistakes or typographical errors. Such data when passed to the algorithm can affect the performance and the results generated. To convert the data to a normalized form, there are three stages of preprocessing; cleaning, integration and transformation.

In the cleaning stage, the data are cleaned by removing missing values and inconsistencies. For the purpose of this project, we are considering only English responses, so French responses were removed. The records of the participants who did not answer or refused to answer were also removed. There were some responses which had

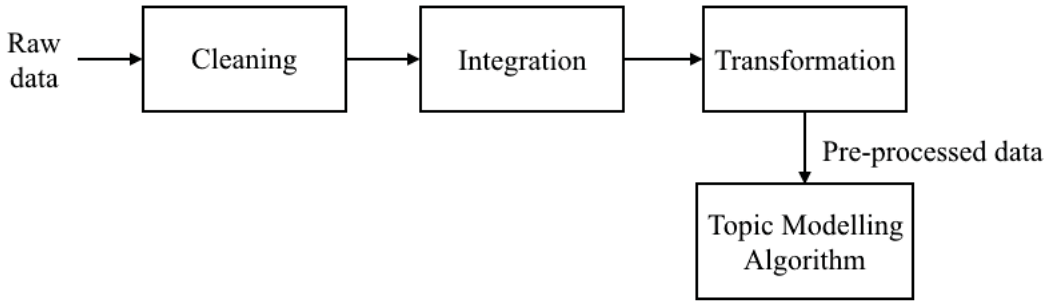

Figure 4.1: Pre-processing

their language tags misplaced. These were addressed and the English ones were considered. Some of the responses had a mixture of both English and French words, the responses were converted to the language from which most of the words were from. The words which were converted to English fully were included. As mentioned before the telephonic and in-home interview data are stored separately. These data were combined and sent to the algorithm. Finally, in the transformation stage, the data are normalized. In certain responses, words were connected by dashes *eg. staying fit-not smoking-moderate consumption of liquor*, when such words were passed on to the algorithm, they were considered as a single word (fit-not) which did not have much meaning. So the dashes were removed to normalize the data. Characters which were not UTF-8 compatible were removed. Punctuation and words which occurred less than five times were removed as well. Now the words are converted to their root form by lemmatization. After all the pre-processing was done, there were 41,500 records which were sent to the module for topic modelling.

Chapter 5

Experiments and Results

This chapter will discuss the evaluation methods used to decide the method which best classifies the documents into topics. The rest of this chapter will compare the results generated by LDA, TKM and our proposed model, CKM.

5.1 Evaluation Method

The authors of the existing topic modelling algorithms have performed their experiments on labelled data. They measured the performance of the model by using precision, recall and F-measure. Since the dataset is unlabelled and it is a time consuming to label all the 50,000 records we looked into the top 20 words and documents assigned to each topic to understand the performance of each model. We took help from experts in epidemiology to help us understand the distinctiveness of each topic as our knowledge on the domain of healthy aging was minimal. Being an unsupervised learning method, interpretation of the results is critical as it varies depending on each individual. Topic modelling algorithms should be able to produce topics which are interpretable by humans. If not the algorithm cannot be considered as a good topic modelling algorithm as the topics generated by it will not make much sense to the users.

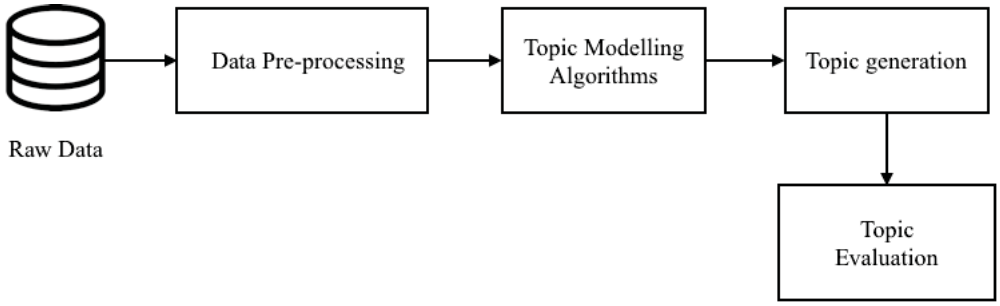

Figure 5.1: Experimental setup

Fig. 5.1 shows the experimental setup used to generate the results which will be discussed later in this chapter. As shown in the setup, first the documents are pre-processed as discussed in chapter 4. For LDA and TKM, during pre-processing, we remove the English stopwords as well, as they are a token-based model. Once the data is pre-processed we get the clean data which is then passed to the topic modelling algorithms. LDA and CKM generate topics based on the input k value whereas, in TKM the model determines the k value.These algorithms give us the top words or n-grams of the topic and the list of documents which belong to each topic.

Using the top 20 words and document list of each topic we are going to compare and discuss the quality of the topic generated by each one of them and find the algorithm which best fits our needs. Experts with epidemiological background helped us in evaluating the results and choose the best model which fits our requirements. The best model should be able to satisfy the following conditions

- The top words or n-grams for a topic should be characteristic to that topic.
- It should be able to produce distinct topics. This means the topics generated by the algorithm should be unique and not have major overlap with any other topic.
- The document distribution among the topic should be fairly uniform. The algorithm should not assign a large number of documents to one topic. This would result in a less number of documents for other topics.
- Since the CLSA dataset is entered by humans, there are a lot of typographical errors or human errors. The algorithm should be robust to such errors.

We also calculated the distance between the documents of a topic, intra-cluster distance as well as the distance between the documents of various topics, inter-cluster distance to see how related the documents are to a topic. The average intra and inter-cluster distance was calculated as mentioned [27]. To calculate the distance between the documents, Jensen-Shannon divergence [29] was used. The distance between the documents d_i and d_j is calculated using the Eq. 5.1.

$$dis(d_i, d_j) = \frac{1}{2} D_{KL}(d_i||m) + \frac{1}{2} D_{KL}(d_j||m) \tag{5.1}$$

This uses Kullback-Leibler divergence [39], where $D_{KL}(p||q) = \sum_i p_i ln\frac{p_i}{q_i}$ and $m = \frac{1}{2}(d_i + d_j)$.

Average Intra-Cluster Distance

$$IntraDis(C) = \frac{1}{K}\sum_{k=1}^{K}\left[\sum_{d_i,d_j \epsilon C_k, i\neq j}\frac{2dis(d_i,d_j)}{|C_k||C_k-1|}\right] \tag{5.2}$$

Average Inter-Cluster Distance

$$InterDis(C) = \frac{1}{K(K-1)}\sum_{c_k,c_{k'}\epsilon C, k\neq k'}^{K}\left[\sum_{d_i \epsilon C_k}\sum_{d_j \epsilon C_{k'}}\frac{2dis(d_i,d_j)}{|C_k||C_{k'}|}\right] \tag{5.3}$$

Ratio between Intra-cluster and Inter-cluster Distances (H score)

$$Hscore = \frac{IntraDis(C)}{InterDis(C)} \tag{5.4}$$

The average intra-cluster distance should be smaller than the average inter-cluster distance meaning the documents belonging to the same topic are closer than the ones in another topic. The intra-cluster distance was calculated for the top 100 documents, which are highly related to the topic or close to the centre point of the cluster. The inter-cluster distance was calculated between the documents of different topics. The ratio between Intra-cluster and Inter-cluster can be calculated using equation 5.4. The topic modelling algorithm which produces distinct topics will have a smaller intra-cluster distance than inter-cluster distance which implies a smaller H score.

Python [40] was used to implement the topic models LDA, TKM and CKM. Numpy [41] and Pandas [42] data frameworks were used to manipulate data. For TKM, most of the code was adapted from [43] which are made available by the authors. For LDA, we used scikit-learn's package [44] which is freely available for use. The experiments were run on hector server available at the Faculty of Computer Science, Dalhousie University. Hector is a multi-core processor with 16 cores CPU.

As mentioned earlier, LDA and CKM get the number of topics, k as an input. So these algorithms were run with different k values 5, 10 and 15. For each k value, we ran the experiment 10 times to assure the consistency of evaluation across the models. The number of topics generated by TKM is pre-determined by the algorithm and this was also run 10 times to ensure the uniformity for evaluation.

5.2 Latent Dirichlet Allocation

For any topic modelling algorithm, the top words produced for each topic should be able to determine the thematic structures present in the corpus. To evaluate the quality of the topics determined by LDA we consider the top 20 words list for topics, $k = 10$. We randomly select the top words list and documents from one of the 10 runs which are shown in table 5.1 and appendix **??** respectively. The words are displayed based on their ranks. This means the first word is more significant to the topic compared to the twentieth word in the top word list. For example, in Topic 1 the first word *happy* adds more significance to the theme of the topic than the last word *community*. By looking into the top words we can identify that some words which are not characteristic of the topic are generated as top words like *good, like, having, just*. The word *good* appears in five topics (topic 1, 4, 5, 6, 9). Though the word implies positivity towards the topics, it is not a word which is specific to a topic and also because it is present in five out of ten topics which means that the word is common to all topics and hence should not be selected as a top word.

Similarly, the word *exercise* is also selected as a top word in five of out 10 topics and at least 9 out of 20 top documents of the all the topics have the word exercise in them. *Exercise* being a domain specific word, is not assigned to one particular topic. If such domain-specific words are very common in the corpus, then it should not be considered as a word which is characteristic to a particular topic. A good topic modelling algorithm should be able to differentiate between the words which are commonly present throughout the corpus and the ones which are distinctive to a topic.

The intra-cluster and inter-cluster distance were calculated using the formula 5.2 and 5.3. The average intra-cluster distance was $3.8638e^{-6}$ and inter-cluster distance

was $8.8941e^{-7}$. The intra-cluster distance is less than the inter-cluster distance showing that the documents between the topics are closely related when compared to documents within the same topic. This shows that topics generated by LDA have overlapping themes and are not unique.

5.3 Token Keyword Model

Unlike LDA, TKM produces its own number of topics for any given corpus. When the pre-processed data is passed to the model, it produces k topics, where the k value is determined by the model. To test the consistency of the algorithm we ran the experiment 10 times. On average the algorithm produced 19 topics. This shows that the algorithm is consistent. Another added advantage of TKM is that the words or tokens are assigned a keyword score based on their frequency and how characteristic they are to a topic. This made the model produce more characteristic top 20 words as shown in table 5.2 and 5.3. Though the top words are meaningful, it can be seen that the model fails to produce distinct topics. By looking at the top words one can identify that more than one topic has similar themes which can be combined. For eg. Topic 7 and Topic 10 both are related to smoking and drinking and contains 1.58% and 0.45% of the documents assigned to it.

In any topic modelling algorithm, the document topic distribution is expected to be uniform. This means that each topic should have a considerable number of documents assigned to them. It was noticed that in every run at least one-third of the topics got less than 1% documents assigned to it. This shows that the document-topic distribution is not uniform. In the run which we have considered for discussion, we found that Topic 3 had no documents assigned to it despite it having top words. Topics 0, 5, 7, 8, 10, 12, 16 had less than 1% of total documents assigned to them.

The average intra-cluster and inter-cluster distance were calculated by using the equations 5.2 and 5.3. The average intra-cluster and inter-cluster distances for all the 10 runs were found to be 0.0207 and 0.0029 respectively. The intra-cluster distance is higher than the inter-cluster distance which shows that the documents within a topic are not aligned to human intuition. The inter-cluster distance is smaller than intra-cluster distance which indicates that the topics have similar themes.

Topics	Top Words
Topic 0	don, eat, moderation, lot, things, drink, smoke, stress, fresh, air, financial, say, vegetables, big, probably, water, worry, able, old, reduce
Topic 1	happy, life, genes, live, involved, happiness, people, home, things, good, family, fit, know, health, makes, money, volunteer, ve, humor, community
Topic 2	active, mind, keeping, busy, body, going, try, like, brain, socially, interested, moderate, weight, learning, looking, better, state, read, time, work
Topic 3	active, keeping, physically, mentally, eating, staying, exercising, properly, things, socializing, busy, new, little, really, hard, young, watch, enjoying, life, work
Topic 4	good, family, friends, activity, physical, mental, social, exercise, diet, having, life, health, sense, habits, work, support, balance, activities, community, stimulation
Topic 5	think, good, food, people, relationships, exercise, nutrition, foods, important, care, just, interaction, time, taking, need, health, eat, things, long, social
Topic 6	attitude, positive, good, exercise, diet, genetics, stress, life, regular, outlook, environment, balanced, living, rest, mental, luck, free, lack, love, laughter
Topic 7	having, activities, doing, social, reading, people, purpose, interests, enjoy, hobbies, faith, regular, outside, strong, connections, look, maintaining, fitness, watching, contact
Topic 8	smoking, exercise, lots, sleep, stay, drinking, getting, eating, alcohol, day, stress, eat, working, low, avoid, away, food, walk, moderation, things
Topic 9	exercise, healthy, eating, diet, good, lifestyle, proper, right, eat, stress, engaged, happy, living, regularly, daily, marriage, level, meals, hobby, plenty

Table 5.1: List of top 20 words generated by LDA for $k = 10$.

Topics	Top Words
Topic 0	good, healthy, relationship, habit, sense, eating, food, sleep, loving, gene, activity, family, exercise, life, nutrition, meaningful, humor, adequate, god, friend
Topic 1	keep, moving, puzzle, mind, sit, going, dont, activity, eat, move, crossword, get, reading, work, time, working, occupied, day, use, try
Topic 2	diet, exercise, stress, genetics, eating, le, balanced, well, low, environment, lifestyle, free, luck, think, healthy, proper, lack, work, eat, living
Topic 3	contact, maintain, others, human, value, generation, loved, younger,wife, support, peer, class, pursuing, aspect, today,playing, safe,working, sport, kid
Topic 4	take, care, seriously, taking, thing, dont, health, life, body, good, doctor, time, stress, medication, eat, people, get, lot, sense, day
Topic 5	forward, look, something, looking, day, health, love, goal, always,purpose, future, going, work, need,content, someone,enough, world, plan, old
Topic 6	social, physical, activity, mental, interaction, stimulation, exercise, family, friend, health, network, contact, engagement, connection, regular, reading, intellectual, nutrition, support, interest
Topic 7	smoke, know, drink, dont, drug, clue, alcohol, lived, much, drank, ive, died, cant, excess, gene, didnt, answer, ag, dad, long
Topic 8	friendship, laughing, fun, laughter, routine, singing, lot, happy, dancing, smiling, et, away, de, love, expectation, exercices, thought, goal, go, stable
Topic 9	thing, new, learning, something, mind, learn, getting, exercise, interest, moderation, involved, open, go, dont, try, always,think, get, friend, live

Table 5.2: List of top 20 words generated by TKM. (Part I)

Topics	Top Words
Topic 10	smoking, drinking, exercise, consumption, avoiding, alcohol, moderate, regular, drug, exercising, quit, moderation, right, much, diet, dietexercise, properly, nutrition, happy, eat
Topic 11	active, keeping, physically, mentally, busy, diet, exercise, eating, lifestyle, healthy, socially, life, attitude, positive, well, mind, activity, social, family, friend
Topic 12	happiness, exercise, family, health, love, nutrition, friend, contentment, general, well, happy, think, laughter, eating, proper, balance, food, money, financial, sleep
Topic 13	eat, fruit, vegetable, food, meat, lot, processed, day, right, exercise, drink, every, walk, sleep, veggie, fresh, protein, much, water, get
Topic 14	people, life, staying, enjoying, happy, around, dont, family, help, think, live, eating, well, enjoy, like, work, child, love, living, get
Topic 15	attitude, positive, exercise, outlook, mental, friend, regular, family, gene, proper, eating, towards, properly, nutrition, humour, thinking, well, health, everything
Topic 16	close, marriage, church, dog, together, ski, job, kid, summer, run, alone, married, extended, grandkids, member, died, read, love, minute, movie
Topic 17	stay, good, diet, exercise, thing, stress, people, healthy, busy, away, eat, smoking, positive, life, social, attitude, physical, food, activity,well

Table 5.3: List of top 20 words generated by TKM. (Part II)

5.4 Character N-gram Keyword Model

The raw data is pre-processed as discussed in chapter 4 and passed to the model. Being a character n-gram based model, we ran experiments with varying number of n-grams ranging from $n = 5$ to $n = 12$ and with k values 5, 10 and 15. Our domain experts helped us in evaluating which value of n yielded the best results. By analyzing the results for various values of n, it was found that $n = 9$ produced better results. For evaluation, we consider one of the 10 runs for $k = 10$ and $n = 9$. The top 20 n-grams produced by that run are shown in tables 5.4 and 5.5.

By looking at the top n-grams and documents for each topic, one can identify that the topics have unique themes. Experts helped us in naming the themes of each topic. For example, "Topic 1: Goals to look forward" means the documents present in topic 1 predominantly have documents which are associated with opinions which talk about having some goals or interests which will keep you going to age well. It was noticed that the topic-document distribution is fairly uniform which means that every topic has got a fair percentage of documents assigned to it, unlike TKM. The model was able to identify variations of the same word and typographical errors. For example, in Topic 6, by looking into the top documents it can be seen that all the words which intents to mean *healthy* but had errors in them are identified. Among the top 20 documents, documents which had the word *healthy* misspell were identified. This shows that CKM is robust to typographical errors.

The average intra-cluster and inter-cluster distance calculated by using formulae 5.2 and 5.3 found to be $1.6813e^{-9}$ and 0.0138 respectively. The difference between the intra-cluster and inter-cluster distance is high showing that the documents between the clusters are distinct from each other. This means that the topics are unique and do not overlap with each other. These show that our model performs better than LDA and TKM.

To show the distinctiveness of any two topics, scattertext [45] was used to visualize the topics as shown in Fig. 5.2. The x and y axes represent the topics 2 and 7 respectively. The words are plotted in space based on their frequency and distinctiveness in the documents belonging to that topic. Both the axes denote the the words which are infrequent towards the intersection and the frequently occurring the words are plotted away from intersection of the axes. The red and blue dots represent topics 2

Topics	Top N-grams
Topic 0	hysical_a, ysical_ac, ical_acti, sical_act, cal_activ, al_activi, ysical_an, sical_and, _vegetabl, vegetable, ical_and_, l_and_men, egetables, cal_and_m, ental_and, ntal_and_, mental_an, l_and_phy, al_and_me, getables_,
Topic 1	mething_t, ething_to, thing_to_, e_and_die, ing_somet, se_and_di, ng_someth, omething_, g_somethi, ving_some, aving_som, something, _somethin, e_to_have, ing_to_do, hing_to_d, _to_look_, _forward_, having_so, _look_for,
Topic 2	_physical, physicall, hysically, _mentally, ysically_, mentally_, y_active_, ly_active, ally_acti, lly_activ, sically_a, ally_and_, nd_mental, _and_ment, and_menta, entally_a, d_physica, _keeping_, nd_physic, _and_phys
Topic 3	lationshi, relations , ationship, _relation, elationsh, friends_, et_exerci, diet_exer, iet_exerc, positive_, tionships, good_diet, _attitude, attitude_, ood_diet_, ionships_, _diet_exe, _yourself, _positive, _properly,
Topic 4	sitive_ou, itive_out, ositive_o, _outlook_, tive_outl, ive_outlo, ve_outloo, utlook_on, tlook_on_, look_on_l, ook_on_li, outlook_o, e_outlook, ok_on_lif, what_you_, k_on_life, hat_you_e, _what_you, _care_of_, at_you_ea,
Topic 5	ing_in_mo, ng_in_mod, g_in_mode, thing_in_, ything_in, rything_i, hing_in_m, erything_, _your_lif, your_life, verything, everythin, n_your_li, in_your_l, hings_in_, _you_are_, that_you_, things_in, _things_i, _that_you
Topic 6	exercise_, _exercise, _healthy_ , ing_activ, xercise_a, ng_active, d_exercis, ercise_an, cise_and_, rcise_and, nd_exerci, _and_exer, and_exerc, g_active_, activity_,,activity, l_activit, t_exercis, lifestyle, healthy_e,
Topic 7	ive_attit, tive_atti, ve_attitu, itive_att, sitive_at, e_attitud, ositive_a, away_from, way_from_, _too_much, _lots_of_, ont_smoke, dont_smok, ttitude_a, a_positiv, nt_smoke_, cessed_fo, od_attitu, ood_attit, too_much_,

Table 5.4: List of top 20 n-grams generated by CKM for each topic for $n = 9$ and $k = 10$ (Part I)

Topics	Top N-grams
Topic 8	eeping_yo, keeping_y, ping_your, eping_you, ng_your_m, ur_mind_a, g_your_mi, mind_acti, _mind_act, ind_activ, r_mind_ac, _your_min, your_mind, our_mind_, p_your_mi, ep_your_m, _and_body, ind_and_b, d_and_bod, nd_and_bo,
Topic 9	_importan, important, mportant_, _involved, _i_think_, s_importa, involved_, _think_th, is_import, i_think_t, _is_impor, y_importa, you_have_, think_tha, hink_that, portant_t, nvolved_i, olved_in_, ry_import, ery_impor,

Table 5.5: List of top 20 n-grams generated by CKM for each topic for $n = 9$ and $k = 10$ (Part II)

and 8 respectively. We can clearly see that there is a line of separation between the words belonging to the two topics.

For example let us consider the word *socially*, it is seen that it occurs 419 times in topic 2 and 4 times in topic 7 as shown in Fig. 5.3. Though certain documents belonging to topic 7 have the word socially, by looking at the documents it is seen that that word is not distinctive to those documents. In document 1, even though the word *socially* occurs, the overall theme of the document is not socially. This shows that our proposed model is able to understand the theme of the documents and assigns it to the appropriate topic.

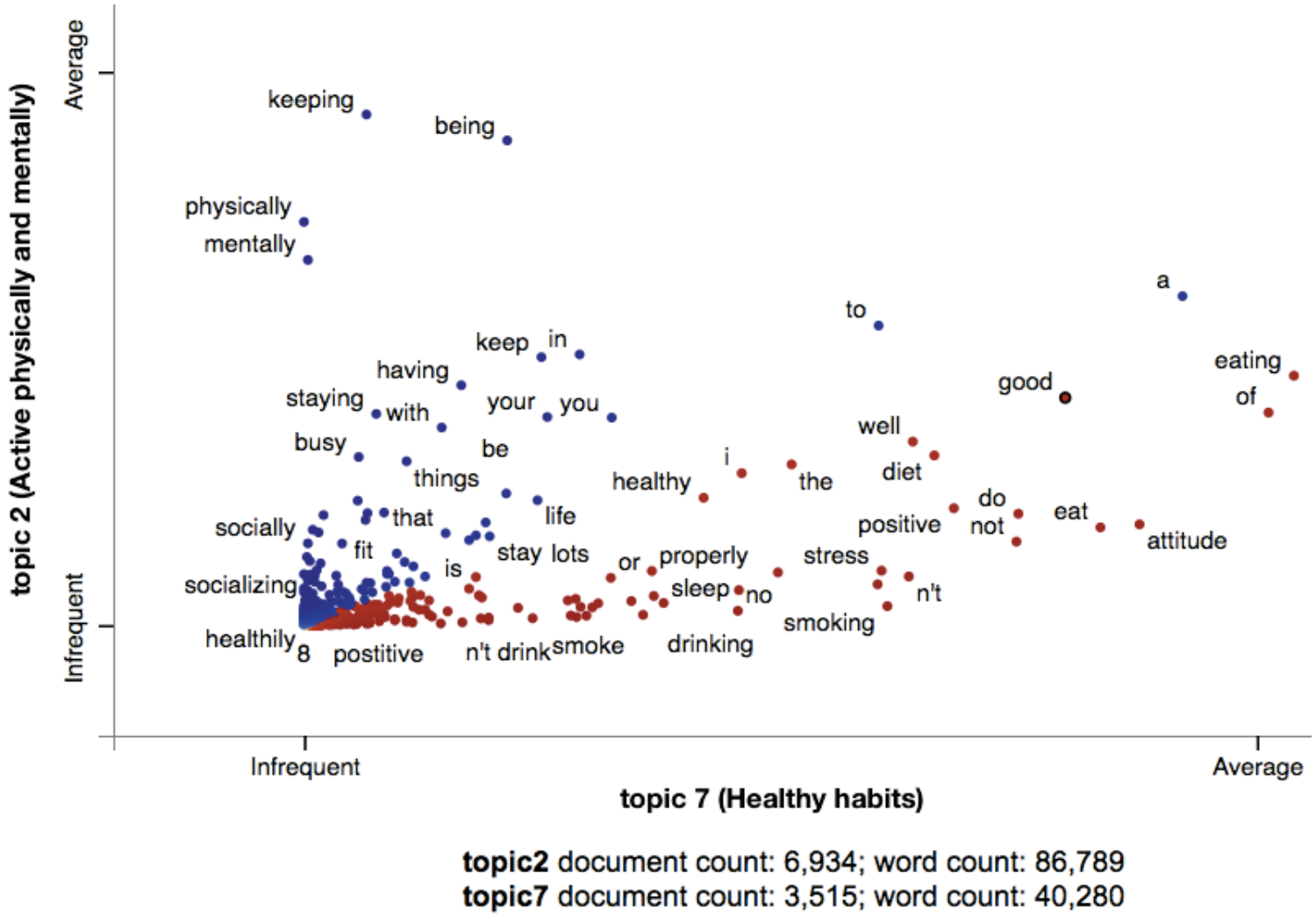

Figure 5.2: Visual representation of Topic 2 and Topic 7

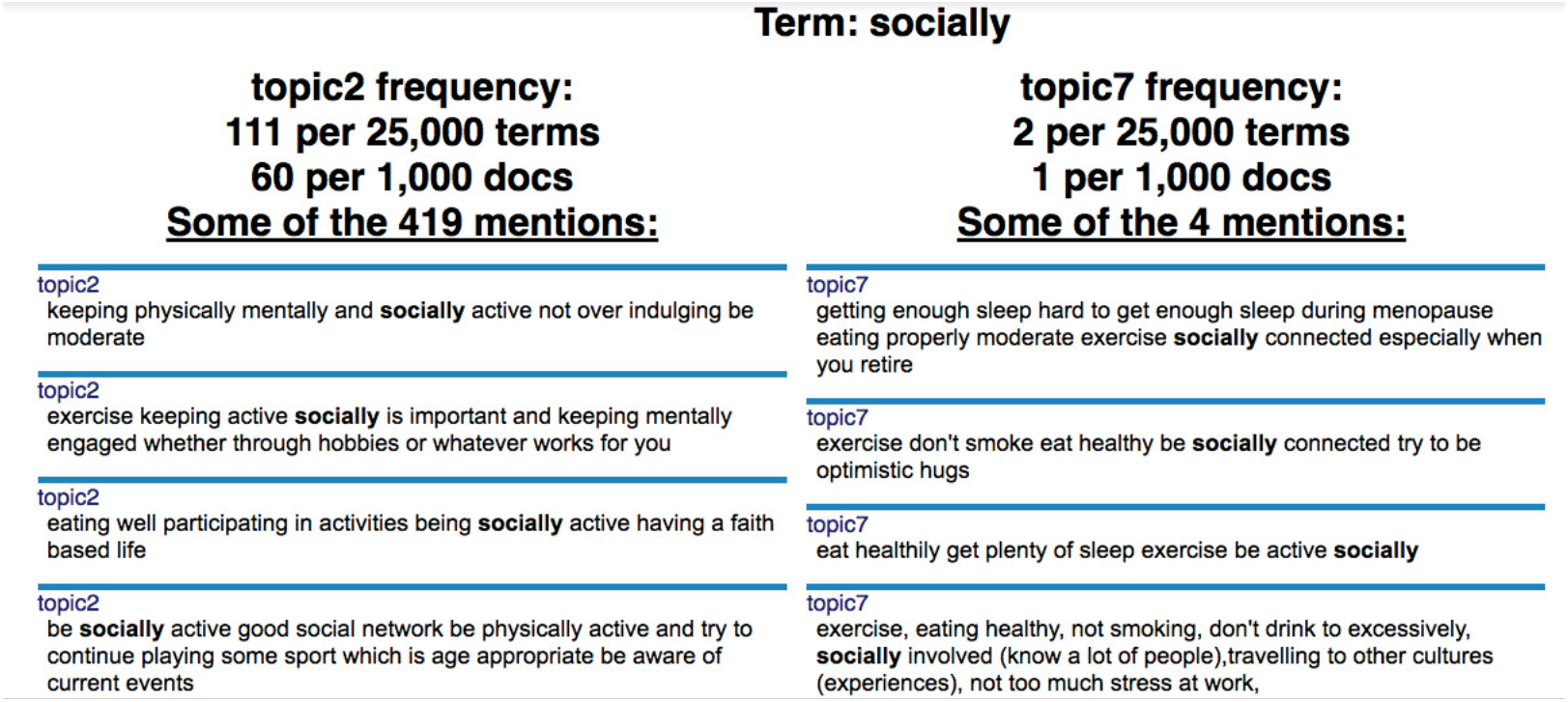

Figure 5.3: Comparison of term socially in Topic 2 and Topic 7

5.5 Discussion

As discussed earlier in this chapter, when looking into the top words/n-grams and documents produced by each model, it is clearly seen that our proposed model, N-gram based Keyword topic model produces topics which align with human intuition. The top n-grams and documents generated for each topic have different theme meaning the topics do not have major overlap between them. It can also be seen that the top n-grams produced by our proposed model is characteristic to the topic unlike LDA. Though TKM also produces characteristic top words, more than one topic has similar top words which results in repetitive topic themes.

Another important factor which needs to be considered to identify a suitable topic modelling algorithm is the distribution of documents in each topic. This means the algorithm should assign a reasonable number of documents to each topic. Though LDA has a good document-topic distribution as shown in Fig. 5.4, the algorithm is not able to differentiate the various themes present in the dataset which makes it not suitable for our dataset. From Fig. 5.5, it is seen that the distribution of documents in TKM is very poor meaning some topics got a few or no documents assigned to it. It was also seen that TKM produced topics which are not distinct. By looking into the Fig. 5.6, it can be observed that our model has a moderate distribution of documents to each topic. Every topic has a decent number of documents assigned to it.

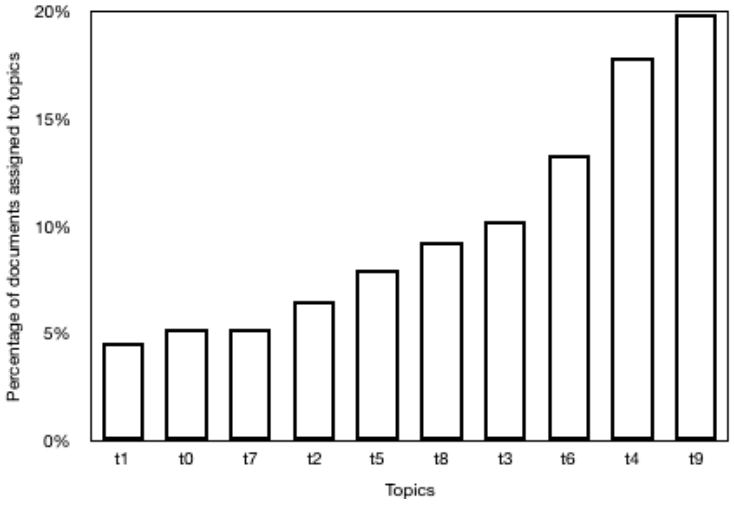

Figure 5.4: Percentage of documents assigned to each topic by LDA with $k = 10$

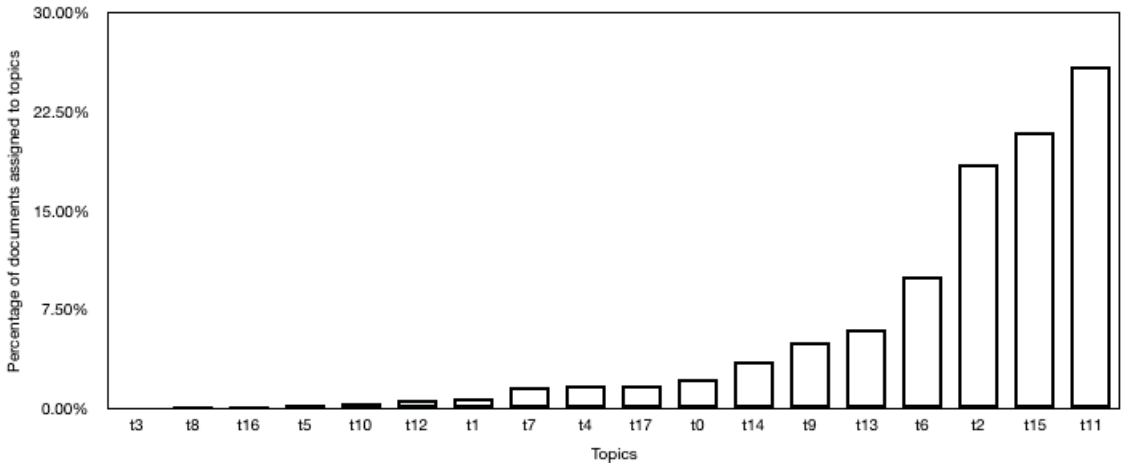

Figure 5.5: Percentage of documents assigned to each topic by TKM

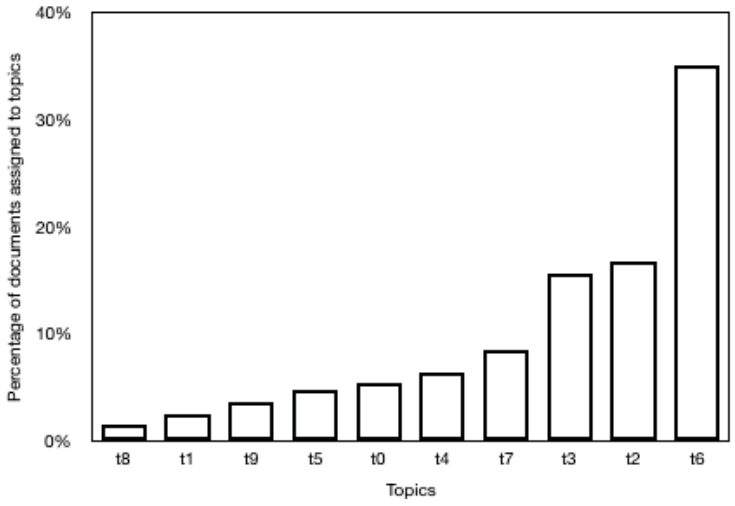

Figure 5.6: Percentage of documents assigned to each topic by CKM with $k = 10$

	Intra-cluster Distance	Inter-cluster Distance	H score
LDA	3.8638E-06	8.8941E-07	3.9345
TKM	0.0207	0.0.0029	10.9712
N-gram	**1.6813E-09**	**0.0138**	**1.2031E-07**

Table 5.6: Average Intra-cluster and Inter-cluster distances and H score for LDA, TKM and CKM models

The intra-cluster and inter-cluster distances are considered to determine how closely the documents are related within the topic and between the topics respectively. The average of intra-cluster and inter-cluster distances and H score for all three models are shown in table 5.6. By looking at the values it is clearly seen that our proposed model has a very low intra-cluster distance of $1.6813e^{-9}$ which expresses that the documents within a topic are closely related. The inter-cluster distance represents the distance between the documents of different topics. Our model has a mean of 0.0138 which is the highest among the models. This describes that the documents belonging to different topics have no or very weak connection between them. The difference between the intra-cluster and inter-cluster distance is very high which also emphasizes that the topics are distinct. The model which has the lowest H score value produces distinct topics or is the best model. Our model has the lowest H score, $1.2031e^{-07}$ proving that it works better than the other ones.

Why LDA does not work well for CLSA dataset:

- LDA considers the document as bag-of-words, where the order of the words in the documents and the order of the documents in the corpus are not considered. It captures the document level word co-occurrences which when applied to short text will make the model suffer from data sparsity problem.

- The words are not repetitive when it comes to a short text document. Since the words present in a document are unique, it makes it difficult for the model to classify words with vague meaning.

- When a word is common in the corpus, it cannot be considered as an attribute to one topic. LDA does not consider the corpus-level word frequencies and fails to recognize the such words.

Why TKM does not work well for CLSA Dataset

- TKM considers the word-topic distribution which is a reason why the model failed to identify various forms of the same word. This leads the model to produce redundant topics.
- Since our dataset has lot of typographical errors, when using tokens the model is not able to identify words which have errors in them. These words are considered as different words. For example, in Topic 10 in table 5.2 it has *dietexercise* even though it has the words separately (*diet* and *exercise*) as top words.

Why CKM works well for CLSA Dataset

- Since the model uses character n-grams, it is robust to typographical errors present in the documents unlike TKM which is token based model. This also the reason why the model is able to identify different forms of the same word.
- Assigning keyword scores to the n-grams emphasis the context of the n-gram and the co-occurrence of words both within the document as well as in the corpus unlike LDA which does not consider the order of words in a document and word co-occurrence patterns in a corpus.
- The average intra-cluster and inter-cluster values indicates that keyword model generates topics which are distinct and make the model a better candidate for CLSA dataset

Limitations of CKM

- CKM produces the top character n-gram for each topic, which are difficult to interpret by the user.
- When the user has a limited domain knowledge, it becomes difficult to identify the number of topics k present in the given dataset.

Chapter 6

Conclusion

In this thesis, we are trying to understand the different topics present in the open ended data collected by Canadian Longitudinal Study on Aging which is focused on the aging process in order to understand the aging process. For this purpose, CLSA dataset comprises of interviews conducted with the elders across Canada to gather their opinion about healthy aging. We are interested in finding the various themes present in their answers to this question on healthy aging. This thesis proposes a novel topic modelling algorithm to identify the themes or topics present in survey data. Data are collected using a questionnaire which consists of close-ended and open-ended questions. The participants choose from a predefined set of answers for a close-ended question which can be converted to graphs or used as data points for algorithms. When it comes to open-ended questions, the participants are allowed to answer in their own words which open to an array of all possibilities. To identify the reoccurring patterns or themes, Grounded Theory's coding techniques were used traditionally. Alternatively, this can be viewed as a topic modelling problem.

Traditional topic modelling algorithms like LSA, PLSA, LDA consider the documents as a BOW model. It means the algorithms do not consider the order of words present in a document. They also ignore the inter documents frequencies which affect their performance when applied to short text. In short text, intra document frequency is almost zero meaning the words are unique. Hence these models suffer from data sparsity issues. To overcome this, TwitterRank [25] suggested increasing the length of each document by collecting tweets from the same user. When there are not enough tweets from a single person, Hong et. al, [26] proposed to collect tweets on the same topic rather than from the same user. These solutions are not always possible, so BTM [27] suggested to consider the word co-occurrence patterns at a corpus-level, unlike the traditional ones which considered it at a document-level. This overcame the data sparsity problem but since the relationship between the words

was not considered, the words which were loosely related and frequently occurred were highlighted.

To overcome all the above issues, TKM [34] suggested which considered the context of the word in the document. This algorithm performed well when the documents were clean and grammatically correct. But generally while taking a survey, participants do not use formal language and are prone to make enough typographical errors. These noises made TKM perform poorly. So we suggest a novel method using character n-grams to solve these issues. CKM is robust to the noise present in survey data as it uses character n-grams which helps in capturing the structure of words which in turn captures the context of words as well. The inter-document frequencies play a vital role when it comes to topic modelling in short text as intra-documents frequencies are almost always zero because practically the words are not repetitive unlike lengthy documents. Using character n-gram data sparsity and the noise in the data is handled effortlessly. By conducting varies experiments, it is seen that our model produces distinct topics or themes when compared to the existing algorithms like LDA and TKM. We calculated the intra-cluster and inter-cluster distances to understand how closely the documents are related within a topic and between topics. The average intra-cluster distance for CKM was less when compared to LDA and TKM. This shows that the documents present in the topics produced by our model are very closely related than the other models. The average inter-cluster distance of CKM was greater than LDA and TKM meaning the documents between the clusters are loosely related and confirms that the topics are distinct.

6.1 Future work

Though our proposed model produces better results than the existing models, there is still room for improvement. One of the major problems of topic modelling algorithms is the input value k, number of topics. It becomes a tedious task to choose k value when the researcher has no or vague domain knowledge. As mentioned before, the TKM model produces its own number of topics but it had a poor performance on our CLSA dataset by producing redundant topics. We would like to explore the reasons why their suggested method failed for short and unstructured informal documents.

CLSA dataset has a lot more details about the participants' socio-demographics like ethnicity, parental background, gender, medical conditions and history, habituals, economic status. We would like to understand the relationship between these socio-demographics and the themes generated by our model. This may give us insights on the phenotype for each theme. Understanding the correlation between the themes and socio-demographics can help us determine the effective ways of healthy aging.

Bibliography

[1] Prarie Research Associates, "A brief history of survey research, Tech note," pp. 1–2, 2008. [Online]. Available: http://www.pra.ca

[2] R. M. Groves, "Three eras of survey research," *Public Opinion Quarterly*, vol. 75, no. 5 SPEC. ISSUE, pp. 861–871, 2011.

[3] E. F. Codd, "A Relational Model of Data for Large Shared Data Banks," *Communications of the ACM*, vol. 13, no. 6, pp. 377–387, 1970.

[4] K. Ashton, "That 'Internet of Things' Thing," *RFiD Journal*, p. 4986, 2009. [Online]. Available: http://www.itrco.jp/libraries/RFIDjournal-That Internet of Things Thing.pdf%5Cnpapers3://publication/uuid/8191C095-0D90-4A17-86B0-550F2F2A6745

[5] C. D. Manning and H. Schütze, *Foundations of Natural Language Processing*. MIT Press, 1999.

[6] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," *IBM Journal of Research and Development*, 1959.

[7] R. Dechter, "Learning While Searching in Constraint-Satisfaction-Problems." *Fifth National Conference on Artificial Intelligence*, 1986.

[8] S. Lohr, "The origins of 'big data': An etymological detective story," Feb 2013. [Online]. Available: https://bits.blogs.nytimes.com/2013/02/01/the-origins-of-big-data-an-etymological-detective-story/

[9] R. Manis, R. Jrgensen, and E. Schaffer, *UNIX relational database management: application development in the UNIX environment*. Prentice Hall, 1988.

[10] J. W. Mohr and P. Bogdanov, "Introduction-Topic models: What they are and why they matter," *Poetics*, 2013.

[11] T. Joachims, "Text Categorization with Suport Vector Machines: Learning with Many Relevant Features," *Proceedings of the 10th European Conference on Machine Learning ECML '98*, 1998.

[12] B. G. Glaser and A. L. Strauss, "The discovery of grounded theory," *International Journal of Qualitative Methods*, vol. 5, pp. 1–10, 1967. [Online]. Available: http://www.ualberta.ca/ iiqm/backissues/5_1/pdf/mills.pdf

[13] M. Sanderson and W. B. Croft, "The History of Information Retrieval Research," *Proceedings of the IEEE*, vol. 100, pp. 1444–1451, 2012.

[14] H. Borko and M. Bernick, "Automatic Document Classification," *Journal of the ACM*, vol. 10, no. 2, pp. 151–162, 1963. [Online]. Available: http://portal.acm.org/citation.cfm?doid=321160.321165

[15] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[16] T. Hofmann, "Probabilistic Latent Semantic Analysis," *Uncertainity in Artifitial Intelligence - UAI'99*, p. 8, 1999.

[17] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *The Journal of Machine Learning Research*, vol. 3, no. 3, pp. 993–1022, 2003.

[18] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.

[19] "CISI and MED dataset," http://www.dataminingresearch.com/index.php/tag/dataset-2, journal=Data.

[20] C. Buckley, "Implementation of the smart information retrieval system," 1985.

[21] E. M. Voorhees, "The Cluster Hypothesis Revisited," *Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval*, pp. 188–196, 1985.

[22] P. W. Foltz, "Latent semantic analysis for text-based research," *Behavior Research Methods, Instruments, and Computers*, vol. 28, no. 2, pp. 197–202, 1996.

[23] T. Hofmann, "Unsupervised Learning from Dyadic Data," *Computational Linguistics*, 1998.

[24] B. de Finetti, *Theory of Probability: A critical introductory treatment.* John Wiley Sons Ltd., Chichester, 1990, vol. 1-2.

[25] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "TwitterRank," in *Proceedings of the third ACM international conference on Web search and data mining - WSDM '10*, 2010, p. 261. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1718487.1718520

[26] L. Hong and B. D. Davison, "Empirical study of topic modeling in Twitter," in *Proceedings of the First Workshop on Social Media Analytics - SOMA '10*, 2010, pp. 80–88. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1964858.1964870

[27] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," *WWW '13 Proceedings of the 22nd international conference on World Wide Web*, pp. 1445–1456, 2013. [Online]. Available: http://dl.acm.org/citation.cfm?id=2488388.2488514

[28] J. S. Liu, "The collapsed gibbs sampler in Bayesian computations with applications to a gene regulation problem," *Journal of the American Statistical Association*, vol. 89, no. 427, pp. 958–966, 1994.

[29] C. E. Shannon, "Prediction and Entropy of Printed English," *Bell System Technical Journal*, 1951.

[30] D. Sundby, "Spelling correction using N-grams," *Technical notes*, 2009. [Online]. Available: http://fileadmin.cs.lth.se/cs/education/EDA171/Reports/2009/david.pdf

[31] Y. Miao, V. Kešelj, and E. Milios, "Document clustering using character N-grams," in *Proceedings of the 14th ACM international conference on Information and knowledge management - CIKM '05*, 2005.

[32] C. Ramisch, "N-gram models for language detection," *Technical Report*, pp. 1–14, 2008.

[33] V. Kešelj, F. Peng, N. Cercone, and C. Thomas, "N-gram-based Author Profiles for Authorship Attribution," *Pacific Association for Computational Linguistics*, 2003.

[34] J. Schneider, "Topic Modeling based on Keywords and Context, SIAM International Conference on Data Mining (SDM)," 2018. [Online]. Available: http://arxiv.org/abs/1710.02650

[35] T. Hofmann, "Unsupervised learning by probabilistic Latent Semantic Analysis,," *Machine Learning, 42(1):177196*, 2001.

[36] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B*, 1977.

[37] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2009.

[38] P. Raina, C. Wolfson, and S. Kirkland, *Canadian Journal on Aging, CLSA Special Issue*, vol. 28, pp. 221–29.

[39] D. Commenges, "Information Theory and Statistics: an overview," no. 1949, pp. 1–22, 2015. [Online]. Available: http://arxiv.org/abs/1511.00860

[40] G. V. Rossum, "Python 2.7," May 2018. [Online]. Available: http://www.python.org/

[41] T. Oliphant, "Numpy," May 2018. [Online]. Available: http://www.numpy.org/

[42] W. McKinney, May 2018. [Online]. Available: http://www.pandas.org/

[43] JohnTailor, "Johntailor/tkm," Feb 2018. [Online]. Available: https://github.com/JohnTailor/tkm

[44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[45] J. S. Kessler, "Scattertext: a Browser-Based Tool for Visualizing how Corpora Differ," 2017. [Online]. Available: http://arxiv.org/abs/1703.00565