USING MACHINE LEARNING AND TEXT MINING ALGORITHMS TO FACILITATE RESEARCH DISCOVERY OF PLANT FOOD METABOLOMICS AND ITS APPLICATION FOR HUMAN HEALTH BENEFIT TARGETS

A Thesis
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Jithin Jose Mathew

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Program:
Genomics and Bioinformatics

November 2020

Fargo, North Dakota

North Dakota State University

Graduate School

Title

USING MACHINE LEARNING AND TEXT MINING ALGORITHMS TO FACILITATE RESEARCH DISCOVERY OF PLANT FOOD METABOLOMICS AND ITS APPLICATION FOR HUMAN HEALTH BENEFIT TARGETS

By

Jithin Jose Mathew

The Supervisory Committee certifies that this ***thesis*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Kalidas Shetty
Chair

Dr. Gursimran Walia
Co-advisor

Dr. Limin Zhang

Dr. Michael J. Christoffers

Approved:

November 23, 2020
Date

Dr. Phillip McClean
Department Chair

ABSTRACT

With the increase in scholarly articles published every day, the need for an automated systematic exploratory literature review tool is rising. With the advance in Text Mining and Machine Learning methods, such data exploratory tools are researched and developed in every scientific domain. This research aims at finding the best keyphrase extraction algorithm and topic modeling algorithm that can aid in Automatic Systematic Literature Review. Based on experimentation on a set of highly relevant scholarly articles published in the domain of food science, graph-based keyphrase extraction algorithms, TopicalPageRank and PositionRank were picked as the best algorithms among 9 keyphrase extraction algorithms for picking domain-specific keywords. Among the two topic modeling algorithms, Latent Dirichlet Assignment (LDA) and Non-zero Matrix Factorization (NMF), the latter performed best in classifying our dataset, which was validated by a domain expert. This research lays the framework for a faster tool development for Systematic Literature Review.

ACKNOWLEDGMENTS

First and foremost, I would like to express my gratitude to my advisor Dr. Kalidas Shetty for this amazing opportunity to pursue my career as a Graduate Research Assistant while pursuing my master's degree at NDSU. His confidence and trust in me played a crucial role in carrying out the research throughout these years. I also like to thank Dr. Gursimran Walia, co-advisor, for his valuable guidance and support in navigating and understanding various topics in the domain of Computer Science. I am grateful to Dr. Dipayan Sarkar, who not only guided me throughout my research but for being a mentor to me throughout my Master's degree.

I would also like to express my gratitude to my committee members Dr. Limin Zhang and Dr. Michael J. Christoffers for their support, advising, and for taking the time to help me finish this research work. I would like to express my gratitude to Dr. Maninder Singh for his valuable support, guidance, and expertise with every aspect of the research.

I would like to thank Jeon, for being an excellent research partner. I would like to thank my family and friends for their support and motivation throughout my career.

I would like to thank almighty God for all the blessings and the right people in my life whom I am always grateful for.

DEDICATION

Dedicated to my family and loved ones who have been encouraging and supportive to pursue my dreams.

TABLE OF CONTENTS

ABSTRACT iii

ACKNOWLEDGMENTS iv

DEDICATION v

LIST OF TABLES ix

LIST OF FIGURES x

LIST OF ABBREVIATIONS xii

1. INTRODUCTION 1

1.1. Problem Statement 1

1.1.1. Proposed Solution 6

1.2. Objectives 7

1.3. Key Terms and Concepts 8

1.3.1. Key Phrase or Term Extraction (KPE). 8

1.3.1.1. Unsupervised Model 8

1.3.1.1.1. Statistical Model 8

1.3.1.1.1.1.Term Frequency Inverse Document Frequency (TFIDF) 8

1.3.1.1.1.2. YAKE 9

1.3.1.1.2. Graph-Based Model 10

1.3.1.1.2.1. MultipartiteRank 10

1.3.1.1.2.2. TopicRank 12

1.3.1.1.2.3. TextRank 13

1.3.1.1.2.4. SingleRank 14

1.3.1.1.2.5. TopicalPageRank 16

1.3.1.1.2.6. PositionRank 17

1.3.1.1.3. Supervised Model 18
1.3.1.1.3.1. WINGUS 18
1.3.1.1.3.2. KEA 19
1.3.2. Semantic Similarity for Topic Modeling 21
1.3.2.1. Latent Dirichlet Allocation 22
1.3.2.2. Non-Negative Matrix Factorization 23
2. BACKGROUND 24
2.1. Systematic Literature Review 24
2.2. Traditional Native American Food 25
2.3. Food Insecurity and Type 2 Diabetes, Obesity Among Native Americans 26
2.4. Keyphrase Extraction from Scientific Articles 27
2.5. Scholarly Data Mining 28
2.5.1. Challenges in Scholarly Data Mining 28
2.6. Topic Modeling and Exploratory Literature Review 29
3. EXPERIMENTATION AND METHODOLOGY 31
3.1. Research Questions 31
3.2. Methodology Used 31
3.3. Preprocessing 31
3.4. Keyphrase Extraction 34
3.4.1. Unsupervised Method 34
3.4.2. Supervised Method 35
3.4.3. Domain Knowledge-Based Keyphrase Extraction 35
3.5. Topic Modeling 36
3.5.1. Latent Dirichlet Allocation 37

3.5.2. Non-Negative Matrix Factorization 38
4. RESULTS AND DISCUSSION 39
4.1. Dataset Analysis 39
4.2. Result Interpretation for Keyphrase Extraction 42
4.3. Result Interpretation for Topic Modeling 56
4.3.1. Latent Dirichlet Allocation 57
4.3.2. Non-Negative Matrix Factorization 61
5. CONCLUSION AND FUTURE SCOPE 66
5.1. Goal 1 - Keyword/Phrase Extraction 66
5.2. Goal 2 - Scraping Data, Indexing and Building a Search Engine 67
5.3. Goal 3 - Toolkit Development 67
REFERENCES 68

LIST OF TABLES

Table Page

1. Corpus of 5 papers, picked by a domain expert for this study 32
2. N-gram distribution in the corpus of 5 papers. 40
3. Confusion matrix summarizing the evaluation method used to count True Positive, True Negative, False Positive, False Negative. 43
4. Difference between stemmed and original keywords. 45
5. Results of automatic keyword extraction using statistical, graph-based, and machine learning based algorithms, D1 represents document 1, similarly, D2, D3, D4 and D5 represent Document2, Document3, Document4, Document5. P denotes Precision, R for Recall, F for F-score. 47
6. Best algorithm performed based on F1-score at picking keywords based on document. 55
7. Table of best performing algorithms based on F1 score for different n-gram. 56
8. List of topics, weights, and keyphrase generated by Latent Dirichlet Allocation. 60
9. Document classification based on the list of keywords generated topic by Latent Dirichlet Allocation. 62
10. List of topics, weights, and keyphrase generated by Non-zero Matrix Factorization. 63
11. Document classification based on the list of keywords generated by Non-zero Matrix Factorization, topics categorized with coherent score described in section 4.2.1 64
12. Best algorithms picked by the study. 65

LIST OF FIGURES

Figure Page

1. Visualization of Topic Modeling Approach (Bird, Klein, and Loper 2009) 22
2. The flow of entire methodology, starting from extracting text from scientific journals to keyphrase/keyword generation (Boudin 2016)..... 32
3. Diagram indicating the topics of interest upon which the corpus of 5 papers fall on. The region indicated in green shows the core concepts of the paper set while the rest of the topics are similar but do not fall into the core research focus 33
4. Workflow depicting the topic modeling and result evaluation..... 37
5. Top 20 unigrams from document 1 before removing the stop words..... 40
6. Top 20 unigrams from document 1 after removing the stop words..... 41
7. Top 20 bigrams from document 1 before removing the stop words..... 41
8. Top 20 bigrams from document 1 after removing the stop words..... 42
9. Visualization of evaluation metrics used to measure the performance of keyword extraction algorithm..... 46
10. Evaluation of algorithms for Document 1, based on Precision, Recall and F1-score for various n-grams..... 51
11. Evaluation of algorithms for Document 2, based on Precision, Recall and F1-score for various n-grams where PositionRank shows best performance..... 52
12. Evaluation of algorithms for Document 3, based on Precision, Recall and F1-score for various n-grams where SingleRank shows the best performance..... 53
13. Evaluation of algorithms for Document 4, based on Precision, Recall and F1-score for various n-grams where MultipartiteRank shows the best performance..... 54
14. Evaluation of algorithms for Document 5, based on Precision, Recall and F1-score for various n-grams where MultipartiteRank shows the best performance for bigrams..... 55
15. Depicts the size of the document, and more importantly the number of keywords picked by the domain expert being directly proportional to the quality of the machine-generated output..... 57

16. Visualization of automatic keyphrase extraction with Multipartite rank algorithm, MeSH, and Plant term database on D4 (Ranilla et al. 2009). The blue highlight indicates the keywords picked by Multipartite Rank, purple for MeSH terms (database assisted), green for common names of plant species, orange for scientific names 58

17. Visualization of automatic keyphrase extraction with TopicalPageRank algorithm, MeSH, and Plant term database on Document D3 (Colby, McDonald, and Adkison 2012). The blue highlight indicates the keywords picked by TopicalPageRank, purple for MeSH terms (database assisted) automatic keyword extraction, green for common names of plant species, orange for fruit names, red is manually highlighted where the keyphrases were not picked 59

18. Distribution 3 different domain-related topics (Native Americans, Traditional food, and Diabetes/health) addressed in 3 topics generated by LDA 62

19. Distribution 3 different domain-related topics (Native Americans, Traditional food and Diabetes/health) addressed in 3 topics generated by NMF 64

20. Visualization of NMF overall term frequency across documents using matrix H and checking which topic has the highest score for each document. The blue bar chart represents the overall term frequency 65

LIST OF ABBREVIATIONS

AI .....Artificial Intelligence.

NDC's .....Non-communicable chronic diseases.

WWW .....World Wide Web.

NLP .....Natural Language Processing.

ML.....Machine Learning.

TF-IDF .....Term Frequency Inverse document frequency.

YAKE .....Yet another Automatic Keyword Extractor.

NER.....Named-entity recognition.

PoS .....Part-Of-Speech.

HAC .....Hierarchical Agglomerative Clustering method.

TPR .....TopicalPageRank.

CP.....Candidate Phrase.

LDA .....Latent Dirichlet Allocation.

NMF .....Non-Negative Matrix Factorization.

PDF .....Portable Document Format.

SVD.....Singular Value Decomposition.

BOW .....Bag of Words.

TP.....True Positive.

FP .....False Positive.

TN .....True Negative.

FN .....False Negative.

LSTM.....Long Short-Memory.

GUI .....Graphical User Interface.

1. INTRODUCTION

This chapter presents the problem statement, dissertation goals, a brief description of key concepts, and the research framework used to derive the objectives and research methodologies for this thesis.

1.1. Problem Statement

Review of literature is an essential and inevitable part of all research irrespective of domains. With the vast amount of data and scientific articles published every day, performing manual literature review is tedious, time-consuming, and comes at the cost of losing valuable domain-specific information. Therefore, automated retrieving, extracting, and structuring of the information makes querying this knowledge easier for researchers (Corney et al. 2004).

Millions of scholarly articles exist on the World Wide Web and databases with valuable information. Searching for highly specific information under a particular domain where the topic of interest falls into a set of focused and intersecting categories is like searching for a needle in the haystack. Still, with the right tool, it is easy to find the correct information within a short period. Systematic Literature Review addresses this issue using two primary techniques, Information Retrieval (IR) and Information Extraction (IE). Information retrieval is used to gather relevant information like an entire set of relevant scholarly articles in Portable Document Format (PDF) addressing a particular topic. Information Extraction (IE) eliminates or reduces the time a researcher has to go through this set of documents by extracting semantic structures, key terms/phrases, concepts, metabolic interactions, and other valuable components, thereby summarizing the critical facts of the document (Corney et al. 2004).

Text mining is gaining popularity among academic and industrial research and development departments. With the unprecedented number of scholarly journals published today,

text mining is becoming crucial in all scientific domains for automating data curation, information retrieval, information extraction, knowledge management, discovery of specialized databases, and tool development (Westergaard et al. 2018). An initiative for open access to scientific knowledge is being promoted worldwide (Lin 2009) making it easy to develop exploratory literature review tools. For example, The National Institute of Health in the United States requires all publications based on their grants to be made available for public access (Lin 2009).

While thousands of scholarly research articles are published on human health challenges such as type 2 diabetes, obesity, and cardiovascular diseases for the wider global population, the information on health disparities and prevalence of non-communicable chronic diseases (NCDs) in indigenous communities such as Native American communities are not widely available and accessible. Additionally, the literature on the impact of dietary changes, especially the loss of traditional food sources on these NCD-linked health disparities of Native American communities of the United States and Canada are even sparsely available in online scientific databases. Overall, NCD is the leading public health challenge causing significant morbidity and mortality, with significantly higher prevalence in indigenous communities worldwide (Kwon et al. 2007). According to the Center for Disease Control and Prevention (CDC), Native Americans are prone to get type 2 diabetes more than any other racial and ethnic groups in the United States (Carter et al. 2008). In addition to genetic, ecological, and economic factors, rapid changes in dietary pattern, especially loss of traditional food sources, have contributed to the rise of type 2 diabetes and other NCDs among the Native American population (Colby, McDonald, and Adkison 2012. ; Mishra et al. 2017). Native Americans also have a higher obesity rate than all other ethnic groups combined in the US (Carter et al. 2008). Data shows that 41% of American Indians and Alaska

Natives are obese (Zamora-Kapoor et al. 2019). Obesity is one of the major health risks commonly associated with a high mortality rate due to cardiovascular diseases, stroke, kidney disease, type 2 diabetes, and cancer (Zamora-Kapoor et al. 2019). While medical research, specifically pharmaceutical drug-based therapeutic interventions have been advanced to address this NCD challenge directly, the health care costs associated with such therapeutic approaches are not always effective for indigenous populations facing higher poverty rates worldwide. In this context, dietary interventions using traditional foods rich in human health protective bioactive compounds are safe and less expensive strategies to prevent and reduce the risks of NCDs in Native American communities. However, the historical epidemiological knowledge to justify the promotion of the traditional Native American diet over modern dietary practices to address the NCD-linked health disparities of Native American communities is hindered by the lack of available data and scientific information (Colby, McDonald, and Adkison 2012). Although type 2 diabetes is a major problem in the vast majority of the population worldwide, the high levels of increase in the numbers among Native Americans in the Northern Plains due to the change in dietary practice has led to an awareness and urgency to scientifically investigate the health benefits of traditional food crops of this region. Since the crops are still grown in a small-scale garden-based farming system, it is never too late to bring back some of this rich traditional food for high-value and health-focused agricultural production. However, for effective integration of traditional Native American food crops for health-relevant dietary and therapeutic interventions, it is essential to understand the specific human health benefits of these traditional food plants. Therefore, the primary aim of this research was to use text mining algorithms and machine learning tools to extract relevant information on the human health benefits of traditional

Native American food plants and to improve our understanding of their specific NCD-linked health benefits (Phillips et al. 2014).

Currently, information regarding traditional plant foods of the Great Plains and their dietary and medicinal uses are not widely available and accessible. In this context, reviewing the knowledge on traditional plant food diversity of the Great Plains and their associated health benefits is essential to address current food and nutritional insecurity linked NCD challenges of the Native American communities. Such revival of the knowledge of the dietary and health benefits of traditional food plants, wild edibles, and other relevant natural food resources are essential for more sustainable and holistic solutions to NCD-linked health disparities of Native American communities (Kwon et al. 2007; Mishra et al. 2017; Ranilla et al. 2019). To achieve this goal, current scientific advances in computational data mining and data analysis can be used to develop widely accessible data resources on Native American plants of the Great Plains and their broader uses in traditional foods and medicines (Sarkar, Walker-Swaney, and Shetty 2019). Such revival of knowledge of traditional plant-based foods of Native Americans and integration of new computational tools based on text mining will contribute to overall strategies to restore and build traditional plant-based food dietary systems for diet and nutrition-linked health benefits of contemporary Native American communities and even further for wider non-indigenous communities of this region and worldwide.

When it comes to the metabolomics of traditional plant-based food, there are minimal resources that provide scientific information on nutritional values and composition (Phillips et al. 2014). The data on food intake patterns among urban American Indians are also very scarce (Carter et al. 2008). Scientific articles on the World Wide Web (WWW) exceed hundreds of millions in the count. Google Scholar alone has an estimate of 100 million publications, making

it extremely challenging to discover useful information. This large corpus of scholarly data is challenging while presenting the opportunities for knowledge-driven discovery (Florescu and Caragea 2017). This research focuses on algorithms that aid in tool development for gathering scientific research information data available online for plant, food, and health science researchers. To achieve this, the first and most crucial step would be to extract keywords from the selected document and use the key phrases to classify the document based on the topics in the corpus. Keyphrase extraction and topic modeling are the two primary techniques of Information Mining and Natural Language Processing. Keyphrase extraction is the process of extracting words or a set of words (sentence) that summarizes the main topic of a document or set of documents. Rabby et al. in 2018 described the quality of a keyphrase with two main terms, popularity, and completeness. A keyword is considered popular if repeated a certain number of times in each document *D*, and completeness is based on the ability of the keyphrase to be interpreted as a whole semantic unit. The extracted keyphrase were used to classify the documents into different categories. Document classification is generally used in the database, and for information retrieval, and machine learning (Rabby et al. 2018). The final output of this project was aimed at selecting the best performing algorithm that can be used for Information Extraction and Text classification. The future goal is to develop a database powered by a local search engine where the database will have a higher information density (Corney et al. 2004) than a regular database with a random set of documents.

1.1.1. Proposed Solution

This study aims at implementing and analyzing advanced computational data mining methods that already exist, and to gather and index a data resource of traditional food crops for the Native American community that will ultimately be used to aid research in health-focused food solutions. This research was aimed at utilizing information mining and extraction of traditional food crops with potential health benefits for Native American communities from a corpus of research journals using Natural Language Processing (NLP) and Machine Learning (ML) and Artificial Intelligence (AI). This research specifically focused on a subfield of NLP, text mining, by employing highly scalable statistical-based techniques to be able to index and search a large corpus of information efficiently. Text mining (an essential component of NLP) can help both when retrieving documents (e.g., primary studies) from search results and when extracting detailed information (i.e., information retrieval) from selected studies. The proposed research was carried out on a small set of highly relevant papers which was further aimed at scaling up a much larger corpus in the future (PubMed) ultimately paving a path to tool development.

The key concepts addressed in this thesis are Keyphrase extraction and topic modeling. Identifying the best Keyphrase extraction algorithm is the first and most crucial step for the following reasons:

1. Gathering more relevant and domain-specific documents by mining databases and the World Wide Web (WWW) using keyword matching and scoring systems.
2. Identify domain-specific entities from the gathered document or database.
3. Summarize the information, identify key concepts, problems and solutions, and create an information-dense database powered by a search engine.

1.2. Objectives

This thesis documents findings from the research conducted to discover the best algorithm for keyword extraction and topic modeling, ultimately helping document classification. This research was aimed at finding scholarly articles for food and health science researchers for automated systematic literature review, which included but is not limited to identifying and picking important key terms (crop names, metabolic terms, bioactive compounds, etc.) and concepts like traditional foods and plants or plant metabolism as described in the title: *"Using Machine Learning and Text Mining Algorithms to Facilitate Research Discovery of Plant Food Metabolomics and its Application for Human Health Benefit Targets."*

The broader goal of this research would be to show that the algorithms and techniques used for the food and health science domains can also be extended to other domains. This research also aims at providing the ability to lay a foundation to build a framework for gathering data for scientific domains with limited data availability.

1.3. Key Terms and Concepts

1.3.1. Key Phrase or Term Extraction (KPE)

KPE is a process of extracting prominent phrases or terms from a NL text with unsupervised and supervised learning algorithms. Research objective in this study is to extract phrases from a NL requiring document by identifying important topics in the text and then extracting prominent phrases from those topics. Keyphrase extraction in this study could be divided into two major classes, Supervised and Unsupervised keyphrase extraction.

1.3.1.1. Unsupervised Model

Unsupervised Learning is a type of machine learning that uses statistical models as an insight to describe hidden structures from unlabeled, uncategorized data (Vermeulen 2019). For this study, unsupervised model is classified into two sections, statistical models, and graph-based models.

1.3.1.1.1. Statistical Model

Automatic keyword extraction is generally performed using statistical models, which include TF-IDF and Expectation-Maximization. Statistical models are also language-independent and are highly reliable when used with simplistic features for general purposes. That being said, adding more features tends to make statistical models more complex to comprehend (Azcarraga, Liu, and Setiono 2012).

1.3.1.1.1.1. Term Frequency Inverse Document Frequency (TFIDF)

Term Frequency Inverse document frequency (TFIDF) uses a function to calculate the inverse proportion frequency of a word in a specific document to the percentage of documents the word appears in and weighs the individual words in a document (Ramos 2003).

Term Frequency (TF) metric evaluates one keyword at a time from all the keywords extracted from N number of documents, and computes its frequency in individual documents d1, d2, d3, d4, and d5. The TF score is calculated by number of times the word appears in each document ÷ total number of words in d1.

Inverse document frequency (IDF) calculates the score of a word by calculating the total number of documents in the corpus divided by the number of documents in which the word appears, then taking the logarithm of that quotient. If the corpus contains N number of documents, and the word w appears in x number of documents, it is calculated as $idf(w) = log(N \div x)$. The inverse document frequency is a measure of how much information the word provides. If the term appears in all the documents, then it is not considered important.

Finally, the term frequency is multiplied by the inverse document frequency, $TF * IDF$ which removes common terms from the document that are irrelevant.

1.3.1.1.1.2. YAKE

YAKE stands for Yet another Automatic Keyword Extractor. The fact that YAKE does not make use of NER or PoS tagger makes it language independent and flexible. The pipeline goes as follows: Preprocessing, feature extraction, scoring, candidate selection, data duplication, and ranking. The features used to determine the keywords are Casing, Word Positional, Word Frequency, Word Relatedness to Context, and Word DifSentence (Campos et al. 2018).

Casing reflects the casing aspect of a word while positional aspect of a feature is based on the observation that relevant keywords often appear at the beginning of a document. Word Frequency, Word Relatedness to Context refers to the number of different terms that occur to the left or right side of the candidate word. Word DifSentence calculates how often a candidate word appears within different sentences.

YAKE combines all the features into a single calculation, assigning a score $S(w)$to each keyword. After generating a set of n-grams, each candidate n-gram is assigned a score $S(kw)$:

$$S(kw) = \frac{\prod_{w \epsilon kw} S(w)}{TF(kw) * (1 + \sum_{w \epsilon kw} S(w))}$$

where a smaller score indicates more significance as a keyphrase, finally, similar key phrases are eliminated using Levenshtein distance where the top ***N*** key phrase list is generated.

While the advantage of statistical models is that they are domain and language-independent, they do not exhibit the performance level of graph-based models.

1.3.1.1.2. Graph-Based Model

Graph-based approaches, while being unsupervised, deliver very good output. The process involves limited steps which can be broken down into: Generating a graph where the nodes represent the words in a document and the edges represents some relationship between them. Proceeding with the first step, there would be a ranking of nodes based on graph-theory measures, finally generating a list of key phrases based on the ranking of the keyphrases (Mihalcea and Tarau 2004).

1.3.1.1.2.1. MultipartiteRank

The multipartite graph represents documents as a set of topic-related candidates. For example, documents 1,7,15 could be three different documents forming a set of a topic while documents 2,5 could be another set of a topic. It also makes use of additional salient features to rank the key phrases rather than just using the semantic relationship of keywords. Keyphrases

occurring at specific parts of the documents are sometimes promoted with edge weight adjustment (Boudin 2018).

The first step involves building a graph of each document where edge weights are used to capture position information. Finally, the ranking algorithm is used to assign a relevance score to each key phrase. Candidate key phrases are selected with pattern matching from the sequence of adjacent nouns with one or more preceding adjectives. Hierarchical agglomerative clustering with average linkage is used to group the stem form of keyphrase into topics. A directed acyclic multipartite graph is built where keyphrases belonging to different topics are connected. Sum of the inverse distance between candidate c_i and c_j is used to weight w_{ij} from node i to j.

$$w_{ij} = \sum_{p_i \epsilon P(c_i)} \sum_{p_i \epsilon P(c_j)} \frac{1}{|p_i - p_j|}$$

where $P(c_i)$ represent the set of word offset positions of candidate c_i. The graph consists of k different independent sets of topics where k represents the number of topics. The first occurring candidate of each document is promoted by adjusting edge weights using the equation:

$$w_{ij} = w_{ij} + \alpha * e^{(\frac{1}{p_i})} * \sum_{c_k \epsilon T(c_j) \setminus \{c_j\}} w_{ki}$$

where $T(c_j)$ represent the set of topics belonging to the topic c_j. p_i represents the offset position of the first occurrence of the candidate c_i. Finally, top N candidates are picked as keyphrases using the TextRank algorithm, which is explained in detail in the following section.

1.3.1.1.2.2. TopicRank

TopicRank uses a graph-based method for keyphrase extraction. Developed as an improvement over the TextRank method, TopicRank represents documents as graphs where candidate keywords are nodes and co-occurrence relations are edges where semantic relationship determines the weight between two nodes. This method uses a complete graph where semantic relationships between topics are captured, while topics are similar keyphrase candidate clusters. Candidate phrases are clustered into topics within a document after text preprocessing followed by ranking the topics within the clusters (document), finally picking one candidate keyphrase for each topic (Bougouin, Boudin, and Daille 2013).

First step involved in topic identification is the extraction of longest nouns and adjective sequences from the document as candidate keyphrases. This step helps to obtain more grammatically correct phrases. Candidate phrases are clustered into a set of topics using Hierarchical Agglomerative Clustering (HAC), where stemmed candidate phrases are considered similar if they meet a minimum overlapping threshold of 25%.

TopicRank uses an undirected graph, while nodes are topics and edges are weighted. Semantic relationships are determined based on how close two key phrases appear in the document. The formula to calculate the weight is:

$$w_{ij} = \sum_{c_i \epsilon\, t_i} \quad \sum_{c_j \epsilon\, t_j} \quad dist\,(c_i c_j)$$

$$dist\,(c_i c_j) = \sum_{p_i \epsilon P(c_i)} \quad \sum_{p_i \epsilon P(c_j)} \quad \frac{1}{|p_i - p_j|}$$

where $dist\ (c_i c_j)$ is the reciprocal distance between $c_i$and c_j, and $P_{(c_i)}$is the offset positions of c_i.

Topic Ranking is performed using TextRank's graph ranking model. A score is assigned to each topic t_i with the formula:

$$S(t_i) = (1 - \lambda) + \lambda * \sum_{t_j \epsilon V_i} \frac{w_{j,i} * S(t_j)}{\sum_{t_j \epsilon V_i} w_{j,k}}$$

where λ is a damping factor and is generally defined as 0.85.

Keyphrase extraction is the final process where most representative candidate keyphrases are selected as keywords. Keyphrases that appear first in the document, most frequently used and the centroid of the cluster appear as keyphrases for the final output.

1.3.1.1.2.3. TextRank

TextRank, originally derived from Google's PageRank, was developed by Brin and page (Brin and Page 1998). The algorithm represents a document as an undirected graph where each node's weight is determined by the information drawn by iterating the entire graph. The weight of a successive or adjacent node determined by the weight of all nodes connecting to them; by logic, the higher the number of nodes connecting to a node, the more significant it is in the graph (Mihalcea and Tarau 2004).

The score of node $V_i$of graph *G* is defined as:

$$S(V_i) = (1 - d) + d * \sum_{j \epsilon In(V_i)} \frac{1}{|Out(V_j)} S(V_j)$$

where *d* is a damping factor usually set to 0.85. The scoring algorithm iterates through the graph after which a score is associated to all the nodes in the graph. Once the graph is built, a ranking algorithm can be used to rank the nodes. The connection between the nodes $V_i$and V_j are measured based on the strength value of edge w_{ij}that connects two nodes. Edge weight is taken into consideration while calculating the score associated with each node in the graph. The following formula is used to achieve the same:

$$WS(V_i) = (1 - d) + d * \sum_{V_j \epsilon In(V_i)} \frac{w_{ij}}{\sum_{V_k \epsilon\, Out(V_j)} w_{jk}} WS(V_j)$$

The output of TextRank is generally configured to retrieve a set of key phrases or words that best represents the natural language document. It uses co-occurrence relation as the lexical unit to determine the potential relationship between two sets of texts. A window of *N* words ranging from 2 to 10 is predetermined as the parameter to define the co-occurrence. A filter is applied earlier to select only lexical units of certain parts of speech (nouns and adjectives). Initially only candidate keywords are added to the graph, which then is reconstructed into keyphrases during post-processing. Weight calculation is performed, and 20-30 iterations are run on the graph with a threshold of 0.0001 until it converges. Top *T* vertices are retained for post-processing where *T* is a fixed value. During post-processing, sequences of adjacent keywords are collapsed into keyphrases generating the final list of keywords.

1.3.1.1.2.4. SingleRank

SingleRank is developed for extracting keyphrases from a single document like a newspaper article (Wan and Xiao 2008). The steps for keyphrase extraction can be divided into

two steps. The first step involves document clustering followed by the next step, keyphrase extraction. Keyphrase extraction is carried out in two different levels, cluster-level keyphrase extraction and document-level keyphrase extraction. Each document d from a set of documents D, contains multiple clusters. A graph *G* is constructed after extracting candidate phrases passing through a rule-based filter. The weights on each edge that connects two nodes $v_i$and $v_j$is computed by finding the co-occurrence of words and distance between the words. A window of 2 to 20 words are set, and words passing through the filter is subjected to the calculation of edge weight using the formula:

$$aff(v_i, v_j) = \sum_{d \epsilon C} count_d(v_i . v_j)$$

A Global Affinity Graph *G* is constructed, which is further represented using an affinity matrix *M*. Once the graph is constructed, the PageRank algorithm is used to calculate the score of each node in the graph.

$$WordScore_{clus}(v_i) = \mu * \sum_{all\ j \neq i} WordScore_{clus}(v_i) * \underline{M}_{j,i} + \frac{(1 - \mu)}{|V|}$$

Candidate phrases ending with nouns and words adjacent to each other are collapsed into phrases. Finally, the entire document cluster is evaluated for nodes (keyphrases) with the highest scores to pick the top *N* keyphrases.

1.3.1.1.2.5. TopicalPageRank

TopicalPageRank (TPR) is a modification of the TextRank algorithm which generates a topic model with word-topic distribution and topic-document distribution. Each word in topics is weighed to generate the most relevant key phrase list that accurately represents a document. The significance of each node is computed using a random walk algorithm that is originally developed from the PageRank algorithm (Sterckx et al. 2015).

The first step of the algorithm extracts topical-candidate keyphrases within a window of 10 words building a directed graph, where nodes represent words and directed edges $w_j$to each word $w_i$represents the edges.

$$R_z(w_i) = \lambda \sum_{j:w_j \to w_i} \left(\frac{e(w_j, w_i)}{O(w_j)} * R_z(w_j)\right) + (1-\lambda) * P_z(w_i)$$

where $R_z(w_i)$ denotes the score for a word $w_i$in topic z. Weight of the edge $w_j$to each word $w_i$is denoted by $e(w_j, w_i)$and λ is the damping factor. $P_z(w_i)$ indicates the probability of word $w_i$occurring in a given topic z.

Once the weights of each word in a topic are calculated, a modified PageRank scoring algorithm is used to score each word w_i in document d rather than calculating the weight of each candidate keyphrase, therefore K topics in D documents, a single weight-value $W(w_i)$. Single-weight value uses cosine similarity between vectors of word-topic probabilities and document-topic probabilities to calculate the single weight $W(w_i)$of word w_i in document d.

$$W(w_i) = \frac{\underline{P}(w_i|\, Z) * \underline{P}(Z|\, d)}{||\underline{P}(w_i|\, Z)|| * ||\underline{P}(Z|\, d)||}$$

The single PageRank $R(w_i)$becomes:

$$R\ (w_i) = \lambda * \sum_{j:w_j \to w_i} (\frac{e(w_j, w_i)}{O(w_j)} * R\ (w_j)) + (1-\lambda) * \frac{W(w_i)}{\sum_{w \epsilon v} W(w)}$$

1.3.1.1.2.6. PositionRank

PositionRank derives its name by its nature of incorporating information from all positions of a word. The co-occurrence of a word is fed to a biased PageRank to score keywords. Ranked keywords are in turn used to rank key phrases, generating a list of keyphrases after post-processing (Florescu and Caragea 2017). PositionRank can be categorized into 3 sections:

1. Construction of word graph
2. Position-biased PageRank
3. Keyphrase selection

The first step involves filtering nouns and adjectives using a part-of-speech filter. The words extracted through the filter are represented as nodes in an undirected graph G. Nodes v_i and v_j are connected by edge $(v_i, v_j)\ \epsilon\ E$. The weight of each edge is computed based on co-occurrences of words within a window *w*.

The second step involves creation of an adjacency matrix *M*. Each node of graph *G* is represented as elements $m_{ij}$of adjacency matrix *M*. If no nodes exist between edges, the value of

v_i and v_j are set to 0. PageRank performs a random walk to compute the weight of a node using the formula (after adding a damping factor):

$$S(t + 1) = \underline{M} * S + (1 - \alpha) * \underline{p}$$

where $\underline{M}$ is the normalized form of matrix M. PositionRank bypasses $\underline{p}$ and assigns higher weights to candidate key phrases found early in the document. The final weights of each vertex are calculated using the formula:

$$S(v_i) = (1 - \alpha) * \underline{p_i} + \alpha * \sum_{v_j \epsilon\, Adj(v_i)} \frac{w_{ji}}{O(v_j)} S(v_j)$$

The final step will be to concatenate words (adjectives and nouns) into phrases of up to a length of three where the weights of individual words are summed to form the score of the keyphrase. At post-processing, the list of keyphrases are sorted by score to generate top *N* keyphrases.

1.3.1.1.3. Supervised Model

Supervised models are pre-trained statistical models that classify keywords or keyphrases using binary classification problems where keywords are either positive keywords or non-keywords (Florescu and Caragea 2017).

1.3.1.1.3.1. WINGUS

WINGUS implements a supervised training model which exploits the logical structure of the document to better extract keyphrases. Although the original WINGUS model was developed

to scrape and gather information from Google Scholar and use a Logical Structure Extraction tool to improve keyword extraction performance. Nguyen and Luong 2010 have only used its Machine Learning based feature extraction and keyphrase generation features (Nguyen and Luong 2010), which has been used in this research.

Naive Bayes classification model is used in WINGUS for candidate selection. Several combinations of features are tested against datasets with different combinations of logical structures. Among the features, *TF*IDF* and First Occurrences are set as the base features for machine learning. Term frequency, term frequency of substring, last occurrences, length of phrases, and other logical structure are some of the other features. After the study, it was shown that features like *TF*IDF*, term frequency of substring, first occurrence of a phrase in a document, and length of phrase, when used in combination, give the best result.

1.3.1.1.3.2. KEA

KEA extracts keywords using Naive Bayes Machine Learning model to classify keywords from text unlike other machine learning methods that use controlled vocabulary to train and test the model.

Preprocessing is carried out where punctuation marks, brackets, and numbers are replaced by phrase boundaries. Apostrophes are removed. Hyphenated words are removed into two and non-token characters are deleted (anything that does not contain letters).

Following preprocessing is candidate phrase identification, where a rule-based candidate selection is performed. Candidate phrases are subjected to stemming to improve the extraction process. The rules for keyphrase extraction are mentioned below.

Rule 1: candidate phrase max length limited to three.

Rule 2: CP cannot be proper names that only ever appear with an initial capital (e.g.: Mary).

Rule 3: CP cannot end or begin with a stop word.

For each candidate phrase appearing in the document, two features, TF*IDF and first occurrence are calculated based on which model classifies the potential candidate phrase as keyphrase or not. First Occurrence is defined as the number of words that precede the phrase's first occurrence. It is divided by the number of words in the document (e.g. if in a document of 100 words, 'Diabetes' occurs first at 21st position, the number of words following diabetes is 79, therefore: 79/100 = 0.79). Important key phrases usually appear at the beginning of the documents. Therefore, these words have higher value under first occurrence calculation which quickly helps to identify critical key phrases (Ian et al. 2005). If any phrase occurs only once in the document, it is discarded, which significantly reduces the data and helps to focus on repeated terms.

During training, a set of authors identified key phrases that are used for training data. For each automatically extracted keyphrase, the following formula is used for computation:

$$P[yes] = \frac{Y}{Y+N} P_{TF*IDF}[t \mid yes] \, P_{distances}[d \mid yes]$$

where *Y* is the number of positive instances, and *N* is the number of negative instances. The probability of a candidate phrase being a positive keyphrase is calculated using:

$$p = P[yes] \,/\, (P[yes] + P[no])$$

During the final post-processing step, keyphrases are sorted based on their score. If two keyphrases have similar score, TF*IDF values of each phrase is used as a tiebreaker. Sub-phrases, which are part of a higher-ranked phrase, are removed from the list, finally returning *N* number of keyphrase.

1.3.2. Semantic Similarity for Topic Modeling

While at the individual document levels, terms/key phrases can be useful, they are not very informative when considering vast amounts of text across different documents. To aid human understanding of a discipline (and its aspects), this research aims to automate the identification of topics (where you can group related words) and identify similar documents. Semantic analysis can help find related/similar text within a document or similar documents. Semantic similarity measures the conceptual similarity between two terms that may not be lexically similar. While there are some resources (e.g., WordNet and MeSH) that can allow researchers to calculate semantic similarity between NL terms and medical terms, respectively, they need to be tailored to a particular domain. Domain experts from Computer Science and Plant Sciences worked together to identify semantically similar terms and tailor our tool that can identify semantically similar terms (and topics) and semantically similar documents to learn and extract "relevant topics" from large documents (Topic Modeling), Latent Dirichlet distributions was used to model each document and each topic, as shown in Figure 1. The output is a ranked list of topics and clustering of papers across different topics based on their probability distribution.

Topic Modeling determines the topic by analyzing a corpus of documents, where a statistical model is used to predict the topic based on a cluster of similar words (Cho 2019). It is

a robust unsupervised analysis of extensive document collections (Anupriya and Karpagavalli 2015; Asmussen, Boye, and Møller 2019).

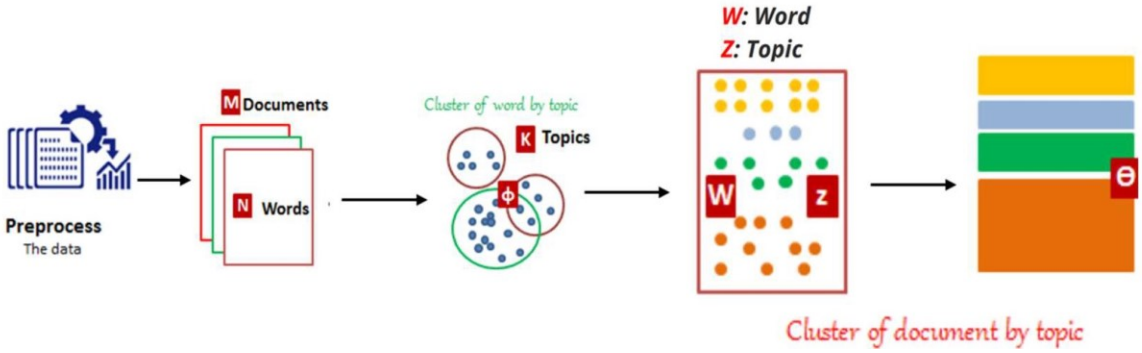

Figure 1. Visualization of Topic Modeling Approach (Bird, Klein, and Loper 2009).

Exploratory literature review, when performed manually, is a time-consuming process with limited output. Even using topic modeling, advanced skills in statistics and computer programming are required to successfully carry out a literature review (Asmussen, Boye, and Møller 2019), therefore developing a tool that helps researchers is the goal of this research.

1.3.2.1. Latent Dirichlet Allocation

LDA is a generative probabilistic three-level hierarchical Bayesian model (Blei, Ng, and Jordan 2003 ; Anupriya and Karpagavalli 2015). A vast majority of scholarly articles claim LDA as a state-of-the-art topic modeling algorithm which provides a quick overview of the main topic addressed in the corpus. A topic is defined as distribution over a fixed vocabulary (Asmussen, Boye, and Møller 2019). LDA being an unsupervised model, determines the topic by analyzing the joint probability distribution between hidden structure of topics and the words appearing in the document. A matrix is created where each element of the matrix represents the probability of each paper being the product of the number of topics by the number of papers (Asmussen, Boye, and Møller 2019).

1.3.2.2. Non-Negative Matrix Factorization

NMF is a non-negative factorization algorithm where a matrix V can be decomposed into two submatrices approximately such that $V = W*H$.

When the rows of V are same as the rows of W , the number of documents, and the columns of V are same as the columns of H as bag-of-words, the columns of W represent main features or topics showing summary of word frequency in each document, while the columns of H indicate which feature or topic that exists in bag-of-words. As a result, the matrix W quantifies the amount of weight of each topic, and the matrix H shows the amount of weight for each topic to a set of bag-of-words.

2. BACKGROUND

2.1. Systematic Literature Review

Systematic Literature Review integrates complex scientific principles into review methodology to summarize scientific research in order to answer prespecified scientific questions using empirical evidence (Thomas et al. 2017). It is mainly used for interpreting published or unpublished data for identifying research gaps and gaining new insights. Some of the main steps involved in systematic literature review are as follows (Jonnalagadda, Goyal, and Huffman 2015):

1. Define a question or problem statement
2. Search for scholarly articles
3. Filter the relevant articles, publication or studies
4. Information/data extraction
5. Evidence Appraisal
6. Meta-analysis
7. Addressing biases

Some of the above-mentioned processes can be performed manually or automatically. For example, the Information Extraction (IE) through NLP saves significant time throughout the review process (Jonnalagadda, Goyal, and Huffman 2015). Text classification (probability of a document to be included in a cluster relevant to the domain of interest) and data extraction (identify relevant information from text) are the two major NLP methods that can be used to automate search, retrieval and IE tasks of systematic literature review (Marshall and Wallace 2019). Marshall and Wallace, 2019 also address the importance of a text classification where keyphrases are classified assessing their relevance to the document to automate literature review.

2.2. Traditional Native American Food

Most of the food crops consumed throughout the world (60%) originated from the North American continent. Some of the popular crops include potatoes, corn, tomatoes, peppers and squash (Park et al. 2016). Aside from the dietary benefits, many of the crops grown in North America were known to be used as medicines while most of the tribe members were known to have basic botanical pharmacy knowledge (Moerman 2008). Currently North America is home to 28,000 different plant species, of which around 2500 of them were used for medicinal purposes by various native tribes (Moerman 2008).

Plant-based foods contain vitamins, minerals, and other bioactive compounds that are health-promoting and disease-preventing (Phillips et al. 2014). Before colonization by Europeans, the native tribes of North America relied entirely on the food crops grown in the region. Wild berries such as serviceberry (*Amelanchier alnifolia*), highbush (*Viburnum trilobum*), chokecherry (*Prunus virginiana*), and buffaloberry (*Shepherdia argentea*) were part of the traditional diet, which are rich in phytochemicals and bioactive compounds that provided health and prevent chronic diseases which are common these days (Burns Kraft et al. 2008). Some of the traditional plant foods include prairie turnips (*Psoralea esculenta Pursh.*), common lambsquarters (*Chenopodium album L.*), cattail broad leaf shoots (*Typha latifolia L.*), stinging nettles (*Urtica dioica L.*), wild plums (*Prunus americana Marshall*), chokecherries (*Prunus virginiana L.*), wild rose hips (*Rosa pratincola Greene*), wild raspberries (*Rubus idaeus L.*), beaked hazelnuts (*Corylus cornuta Marshall*), and plains prickly pears (*Opuntia polyacantha Haw.*) (Phillips et al. 2014).

However, similar to how the rest of the world has adopted food crops, traditional Native American food practices underwent changes by the intervention of food crops from all over the

world. Research today points out that these changes in food habits also have had several negative impacts on the health of Native American communities.

2.3. Food Insecurity and Type 2 Diabetes, Obesity Among Native Americans

Food and nutrition-related health disparities has been recognized as a severe problem for Native American communities especially those living in tribal land (Pindus and Hafford 2019). For the past half-decade, diabetes has had a dangerous role in the morbidity and mortality of Native American Indians, out of which 95% is due to type 2 diabetes alone (Patchell and Edwards 2014). During the dynamic growth of the United States, the shrinkage of land and moving further away from their ancestral land where traditional food crops were once grown, the American Indians relied more on commercially available canned food and less on traditional food (Patchell and Edwards 2014).

Comparing other racial and ethnic groups, American Indians have higher rates of obesity, and diabetes rates due to high calorie, low nutritional value food consumption. Surveys show that, in combination with unhealthy dietary practices, higher smoking rates increase their risk of cardiovascular disease by many folds. Native Americans in the Northern Plains have 58 % higher mortality rate due to heart disease compared to the white population (Warne and Wescott 2019) and some studies also show that 80% of the tribal elders have diabetes (Patchell and Edwards 2014).

Treating diabetes with medication is the first half of the solution. Addressing the problem by changing the root causes, by shifting the focus to addressing lifestyle and nutritional issues will benefit the native community in the long run. Although type 2 diabetes is a significant problem in the vast majority of the population worldwide, the increase in the numbers among

Native Americans in the Northern Plains due to the change in dietary practice has led to an awareness and urgency to research the health benefits of native food crops.

2.4. Keyphrase Extraction from Scientific Articles

Keyphrase extraction is the fundamental task where a set keyphrase is mapped to a document representing the concept (Mahata et al. 2018). Once extracted, the keyphrases can be used for a document retrieval system for search engines, classifying, and clustering documents into databases. Keyphrase extraction algorithms work on a scoring system where a set of potential candidate phrases are picked from a document by heuristic rules followed by ranking each keyphrase based on their relevance to the document which again is determined by a set of rules (Tokala Yaswanth Sri Sai et al. 2020).

Keyphrase extraction is primarily classified into four main categories: linguistic approaches, machine learning approaches, domain-specific approaches, and statistical approaches. Furthermore, Machine Learning-based systems could be further categorized into supervised approach, unsupervised approach, and semi-supervised approach (Rabby et al. 2018). Among various keyword extraction techniques, Graph-based ranking algorithms (unsupervised learning) like TextRank, TopicalPageRank have displayed promising and reliable results by deriving knowledge from the entire text. Supervised learning approaches classify the keywords using binary classification, therefore the keywords either fall into one of the two categories, keyword or not a keyword. However, the requirement of large training data is a limiting factor, while the results are not up to the mark.

Statistical models like TFIDF, YAKE, are domain and language-independent models, which do not require training data while still exhibiting reasonable performance during classification. To extract most relevant keywords, semantics, contextual and grammatical

relations need to be taken into consideration which the statistical models are not designed to do (Rabby et al. 2018).

2.5. Scholarly Data Mining

The number of scientific articles a human can read a day, week or even month is limited. While scientific research involves a tremendous amount of literature review and analysis, tools to aid a researcher is limited. It is reported that a scientific article is published every 20 seconds (Saggion, Horacio, and Ronzano 2017). PubMed (https://pubmed.ncbi.nlm.nih.gov/), an online repository of biomedical literature from MEDLINE, alone contains more than 30 million published scholarly articles. The rate of free online publications is also growing at an unprecedented rate. About 17% of Scopus and ISI Web of Knowledge indexed is accessible to users (Saggion, Horacio, and Ronzano 2017). While the scholarly data is aggregating throughout the World Wide Web (WWW), the field of Natural Language Processing and its advances in analyzing scientific information presents limitless opportunities to the scientific community.

While we have the opportunity of mining through this huge amount of digital information, there are also several challenges to this task of which some of them are addressed below:

2.5.1. Challenges in Scholarly Data Mining

1. The first challenge is to identify the right article which requires keyword/phrase extraction followed by classification of the document while handling and extracting text from Portable Document Format (PDF) formats (Duy Duc AnDD, Fiol, and Jonnalagadda 2016).

2. Articles are highly structured, which is suitable for manual reading, although, from a text mining perspective the difference in the structure and formatting of text is a hurdle in navigating through the data. In such a situation, “one rule fits all” do not work here.
3. Extraction Semantics and Contextual relations from an article remain a challenge (Adnan and Akbar 2019).
4. Extraction of scientific names, chemical compounds, key terms, their interaction with each other, and its overall relevance to the topic of interest is a difficult task (Saggion, Horacio, and Ronzano 2017).
5. Availability of the data, easy access is curtailed to data mining. Creation of datasets for training and testing various machine learning algorithms is time-consuming. While some domains like computer science are well saturated with labeled and structured free-accessible datasets, other domains lack the direct availability of datasets.

2.6. Topic Modeling and Exploratory Literature Review

Exploratory literature review is a time-consuming process when performed manually (Asmussen, Boye, and Møller 2019). Classification of documents into sections and catalogs dates back to the early centuries when paper-based information started to pile up. Traditionally classification of documents is carried out by experienced individuals with domain knowledge, which is labor-intensive and time-consuming. Topic modeling presents excellent research opportunities for exploratory and literature review (Asmussen, Boye, and Møller 2019).

Topic Modelling is a statistical modeling technique used for automatically finding the topic based on the content of the data corpus and classifying individual documents into respective classes (Tong and Zhang 2016). Asmussen et.al 2019 in their work, used Latent

Dirichlet Allocation (LDA), one of the most widely used methods for exploratory literature review.

3. EXPRIMENTATION AND METHODOLOGY

3.1. Research Questions

This research attempts to find an answer to the following key questions:

1. What is the best performing keyword/phrase extraction algorithms for literature focused on the health benefits of traditional foods?
2. How can the output from keyphrase extraction methods be used to identify topics of interest from a large corpus of literature?

3.2. Methodology Used

Keyword extraction performed here can be classified using two methods, Supervised and Unsupervised. Supervised classification approaches use algorithms that need direct or indirect human intervention for learning and classification. Most of the keyword extraction is performed using PKE: an open-source python-based keyphrase extraction toolkit (Boudin 2016) while database-assisted, hybrid methods of keyword extraction have also been experimented with.

Five scientific articles addressing a diverse set of topics (Figure 3, Table 1) in our area of interest were selected as the dataset for the analysis, explained in the upcoming sections. Each file is subjected to Automatic keyword extraction using Supervised and Unsupervised methods and finally comparing it to lists of handpicked keywords by a domain expert for individual papers.

3.3. Preprocessing

Preprocessing is a crucial step in text mining, which has a tremendous impact on the output. General preprocessing methods include tokenization, stop-word removal, lower case conversion, and stemming (Uysal, Kursat, and Gunal 2014). Although these methods alone hold

good for most machine learning techniques, to remove misspelled and unknown words, additional preprocessing needs to be carried out to address this.

Table 1. Corpus of 5 papers, picked by a domain expert for this study.

No	Papers
1	Social Determinants of American Indian Nutritional Health (Warne and Wescott 2019)
2	Food Diversity and Indigenous Food Systems to Combat Diet-Linked Chronic Diseases (Sarkar, Walker-Swaney, and Shetty 2019)
3	Traditional Native American Foods: Stories from Northern Plains Elders (Colby, McDonald, and Adkison 2012)
4	Evaluation of Indigenous Grains from the Peruvian Andean Region for Antidiabetes and Antihypertension Potential Using In Vitro Methods (Lena Galvez et al. 2009)
5	Dietary Change and Traditional Food Systems of Indigenous Peoples (Kuhnlein and Receveur 1996)

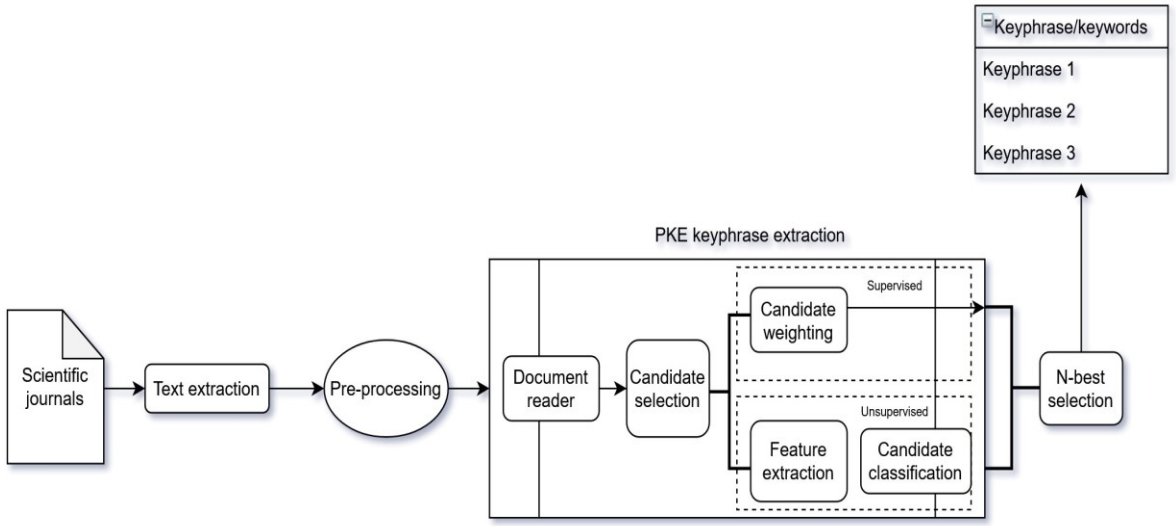

Figure 2. The flow of entire methodology, starting from extracting text from scientific journals to keyphrase/keyword generation (Boudin 2016).

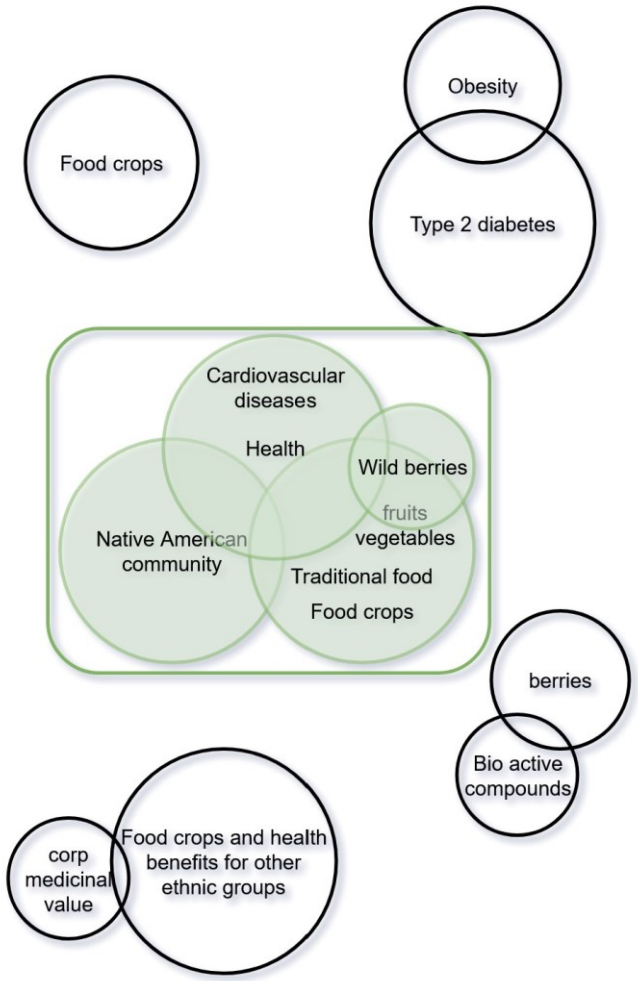

Figure 3. Diagram indicating the topics of interest upon which the corpus of 5 papers fall on. The region indicated in green shows the core concepts of the paper set while the rest of the topics are similar but do not fall into the core research focus.

Research papers published are usually available in the Portable Document Format (PDF) format. Although PDFs are the most widely used formats by different publishers to present their papers, it is not suitable for direct text mining applications (Corney et al. 2004). Therefore, text from each PDF is extracted and preprocessed to improve the keyword/keyphrase extraction process and reduce false positives. Alpha numerals within the parentheses, along with email addresses and URLs are removed. Hyphens are removed, and the words are joined, and some of the special characters are removed. Non-English characters/words and single characters are

removed with regular expressions. Part-of-speech tagging (PoS tagging) is carried out during which verbs are extracted from the document

Keyphrase segmentation: Due to the errors in text extraction from PDFs, two or more keywords are joined creating a misspelled word or a nonsense word. For example, “berrysized” is expected to be extracted as separate keywords ‘berry’ and ‘sized’ which can be classified as misspelled. While some keywords are removed to help keyphrase/keyword extraction algorithms to perform with better accuracy, others are subjected to further validation across medical and scientific term databases, and spell check is performed.

3.4. Keyphrase Extraction

Keyphrase Extraction is carried out using the PKE library, a Python-based Keyphrase Extraction algorithm (Boudin 2016). The first step toward keyphrase extraction is extracting text from the Portable Document Files (PDFs). The extracted text is preprocessed to improve the extraction results. There are no ways to measure the quality of the processed files. Therefore, once the files are saved, they are directly subjected to keyword extraction algorithms.

The keyphrase extraction is mainly classified into two classes, unsupervised method, and supervised method.

3.4.1. Unsupervised Method

Unsupervised methods are further divided into statistical methodology and graph-based methods. Statistical models include TF-IDF and YAKE. Graph-based models consist of MultipartieRank, PositionRank, TextRank, SingleRank, and TopicalPageRank.

Each algorithm works by picking a list of words from an individual document called candidate selection. The unsupervised method works by weighing each candidate phrase based on a set of rules. While statistical models used some statistical formulas, as mentioned in the

introduction section, to weigh each candidate. Graph-based models convert each document into a graph with nodes and edges, and the connections determine the weight of each candidate keyphrase. Once the graph is parsed through, top n-best keyphrases set by the user are returned.

3.4.2. Supervised Method

Supervised methods used in this research include KEA and WINGUS. Supervised methods extract keyphrases based on features upon which it is trained. KEA and WINGUS are both trained by a set of key phrases called the gold-annotation keyphrase handpicked by the domain expert. Once a series of documents relevant to food science is saved as a corpus of text files, both the gold-annotations and text documents are passed as input to the training algorithm. During the training, the weights are saved as a Python picked file which is called during keyphrase extraction.

Initial steps of the keyphrase extraction resemble the Unsupervised method, where each document is parsed to generate a list of candidate keyphrases. The features of candidate phrases are extracted and passed through the loaded model file. During the final step, if the candidate phrase is deemed to fit as a key phrase, it is returned as a true positive. Overall, the top *n* keyphrase set by the user is returned.

3.4.3. Domain Knowledge-Based Keyphrase Extraction

Although generally the keyphrase extraction is broadly classified into Supervised and Unsupervised methods, some studies with the ready availability of domain-specific keyphrase databases like MeSH use a hybrid model for keyphrase extraction, which tend to increase the overall performance of the keyword extraction. For this research MeSH (Medical Subject Headings) (Lipscomb 2000), a database of domain-specific vocabulary hosted and maintained by the U.S. National Library of Medicine, and a plant name database obtained from the USDA

Plants Database (United States Department of Agriculture) were used. The objective here is to identify and pick specific medical terms that are associated with cardiovascular diseases and crop names that have potential health benefits. Crop names were further separated into common names and scientific names. The count of scientific terms used in this research, including all crops grown in North America alone, is approximately 90748, while common names had 4988 entries. There were around 12,324 entries of medical subheadings in the MeSH database after removing terms that were not domain specific. Trie search (Willard 1984) algorithm was implemented to search keywords faster from multiple databases to save time.

Trie search reads the list of phrases from above-mentioned databases into Trie entries. Once the Trie entries are complete, a PDF file is created from which the text is extracted and subjected to pre-processing (explained in section 3.3). A set of N-grams, starting with unigram to 4-grams are generated, and each list is searched against the Trie entries using Trie search. If the search has an exact match with a term or phrase in the existing database, the result is stored and returned. A visualization of the search results can be seen in figures 16 and 17.

3.5. Topic Modeling

Most topic modeling techniques depend on the quality of text preprocessing and the optimal determination of the number of topics. As a set of key phrases are prepared from the expert, only sentences from the PDF files that include those phrases are analyzed as an input source. Also, the SVD (Singular Value Decomposition) algorithm is chosen to find the number of optimal topics. The entire workflow of topic modeling involved in this project is shown in figure 4. Topic modeling outputs are generated as:

1. Keyword-wise topic representation
2. Topic-wise document representation

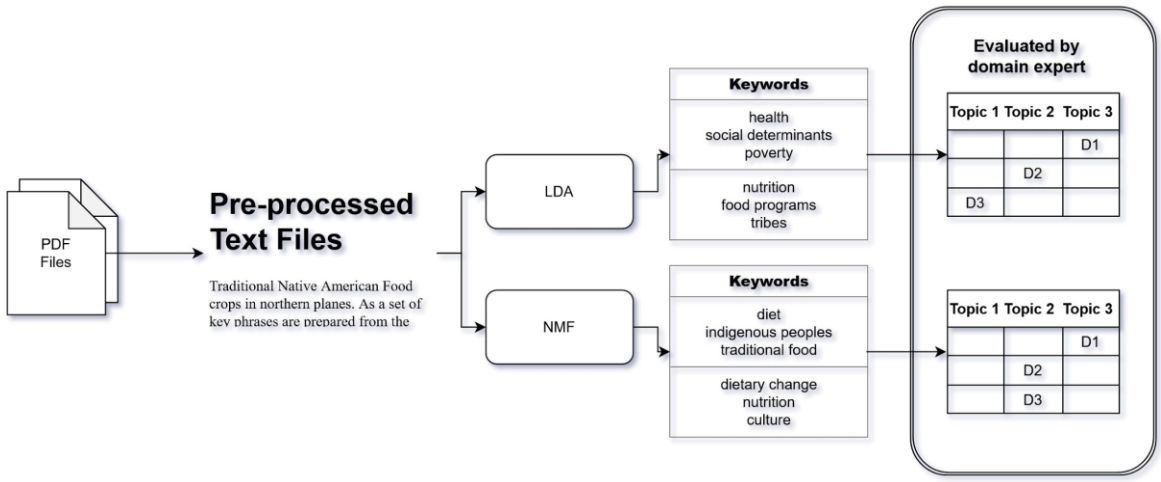

Figure 4. Workflow depicting the topic modeling and result evaluation.

3.5.1. Latent Dirichlet Allocation

LDA (Latent Dirichlet Allocation) is a powerful unsupervised technique to extract relevant topics quickly and easily from a large set of text. Python's Gensim package is used to perform topic modeling in this research. A document term matrix is generated to decide the list of keywords aligned with the key phrases from the expert and calculate word frequencies. Although Gensim's LDA can build the bigrams, trigrams, and other n-grams, they were not used here. After initializing and training the LDA model, a list of words by each topic and topic weight by each document is returned.

The workflow begins with preprocessing, which is already performed for keyphrase extraction where each extracted and processed text is saved as documents of a small corpus. For additional improvement in the quality of the results, the words are stemmed to their root form using porter-stemming technique. Each document is then analyzed to find the words and the number of times the word appears in the document, popularly known as bag-of-words (BOW). Following the generation of BOW, the Term-Frequency, and Inverse Document Frequency

(TFIDF) is created for each word in BOW resulting in a dictionary of terms and its frequency. The LDA model is run separately with both BOW and TF IDF data. The Gensim parameters for BOW is set to: number of topics = 3, passes = 3000 and iterations = 400 for running the LDA algorithm. For TFIDF, the parameters were predefined as topics = 3, passes = 2 and workers = 4. Once the topics are generated and the documents are classified into the most suitable topic, the output is subjected to evaluation by a domain expert.

3.5.2. Non-Negative Matrix Factorization

NMF is a matrix decomposition algorithm that finds two target matrices from a large source matrix showing a weighted sum of topics. It is an unsupervised matrix factorization method known for incorporating simultaneous dimensionality reduction and clustering (Albalawi, Yeap, and Benyoucef 2020). For this research, an NMF module from the Scikit-learn package is used. The same text information is used in the LDA and is passed into the algorithm as input texts for the topic analysis, where the same format of results set with different outcomes is observed.

Like LDA, the preprocessed text files were passed into the NMF algorithm with input as Term-Document matrix and number of topics. The Term-Document matrix is generated using the function TfidfVectorizer from the Sklearn's text feature extraction module. Once the Term-Document matrix is obtained, it is passed to NMF algorithm, with the number of topics set to three as a parameter.

4. RESULTS AND DISCUSSION

4.1. Dataset Analysis

The five research articles used for this research are related to traditional food and its health impact on Native American communities. Since the main goal of this work is to find scientific research articles in food science that address the health impact and benefits on Native American communities, the keywords/key phrases can fall into one of the two categories,

i) Keywords/keyphrases relevant to the topic

ii) Not relevant to the topic

A summary of our dataset is given in table 2. Words like ‘health,’ ,‘nutrition,’ ‘blueberries’, and phrases like ‘traditional food,’ and ‘Native American’ were picked by domain experts. Once the keywords were marked on the selected scientific research article, a list of keywords and phrases are generated. The list is subjected to porter stemming (Table 4) to remove mismatches caused by suffixes. For example, the term ‘interventions’ or ‘intervention’ becomes ‘intervent’ to get the best match with the algorithm generated keywords, which are also subjected to stemming. Each of the papers is subjected to preprocessing. Preprocessing is an essential step in keyword extraction, and Figure 5 shows the significance of preprocessing. For example, a quick analysis of unigrams before removing stop words shows that words like ‘and’, ‘of’, and ‘the’ appear in significantly higher quantity. The five documents that we analyzed with a total of 27,634 words have a combined count of 3,714 stop words like ‘and,’ ‘of,’ and ‘the’ in the corpus. Keeping the stop words alters the results significantly in a negative manner. Figure 5-8 shows the top unigrams and bigrams before and after removing the stop words.

A quick review of document D1 (Figure 5) shows that words like ‘and’, ‘of’ or other stop words appears between 600-1300 times. Reviewing the document after removing the stop words

(Figure 5) also shows that important keywords like ‘food’, and ‘traditional’ appear the same number of times. Subjecting the papers to text mining algorithms before preprocessing will significantly reduce the quality of the generated output.

Table 2. N-gram distribution in the corpus of 5 papers.

Paper	No. of words	No. of unique words	2-grams	3-grams	4-grams
Paper 1	4211	1263	3096	3096	3095
Paper 2	5422	1227	3409	3409	3409
Paper 3	3615	1007	2625	2624	2623
Paper 4	5784	1495	3859	3858	3857
Paper 5	8602	1989	5852	5852	5852

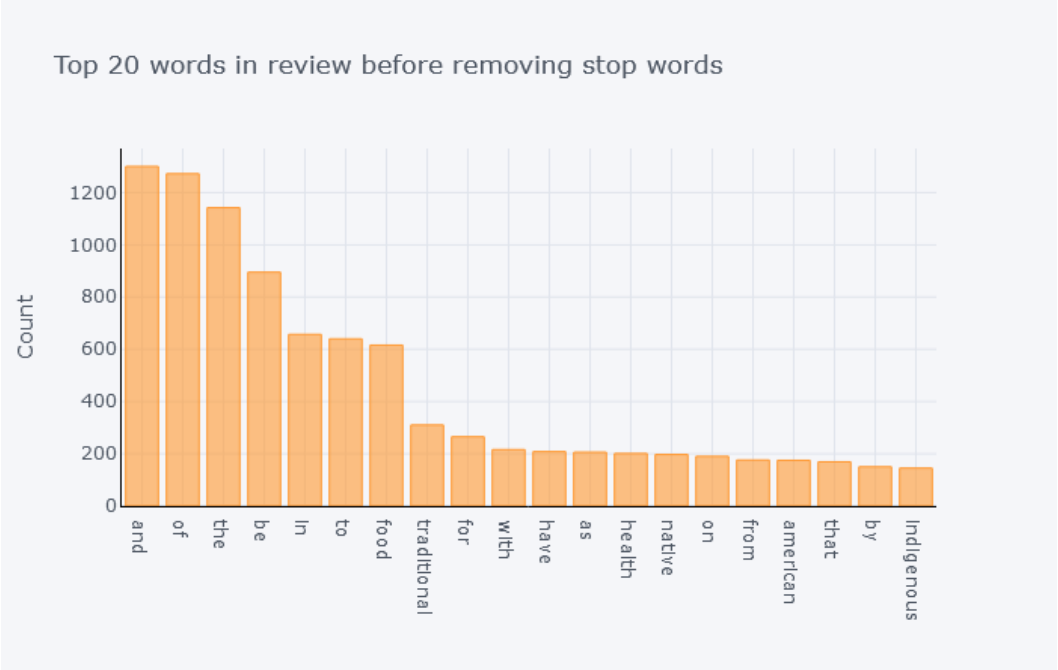

Figure 5. Top 20 unigrams from document 1 before removing the stop words.

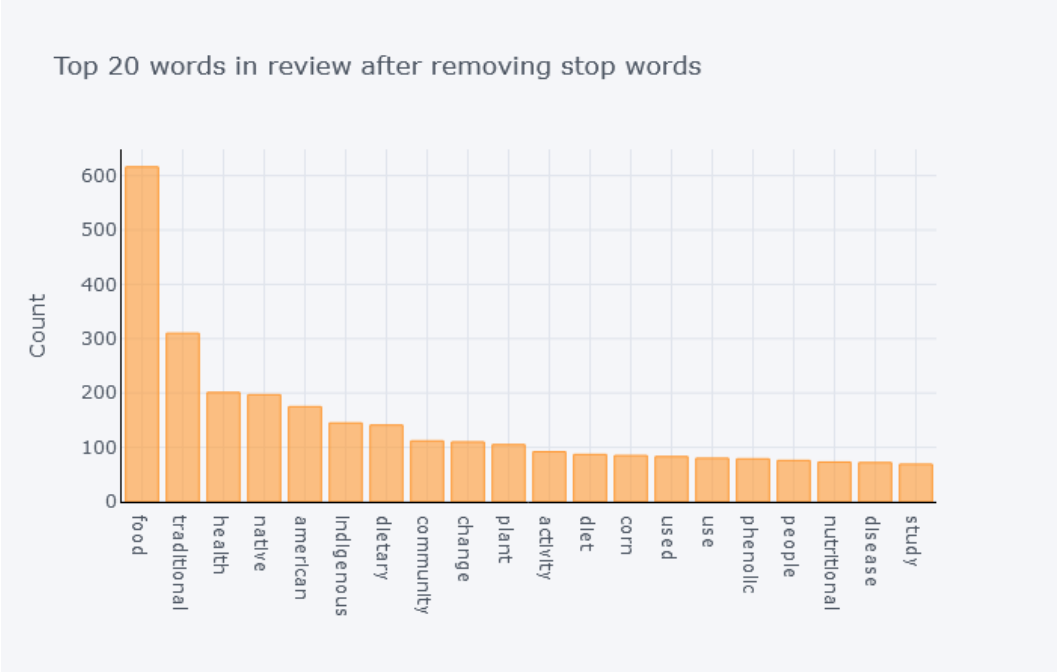

Figure 6. Top 20 unigrams from document 1 after removing the stop words.

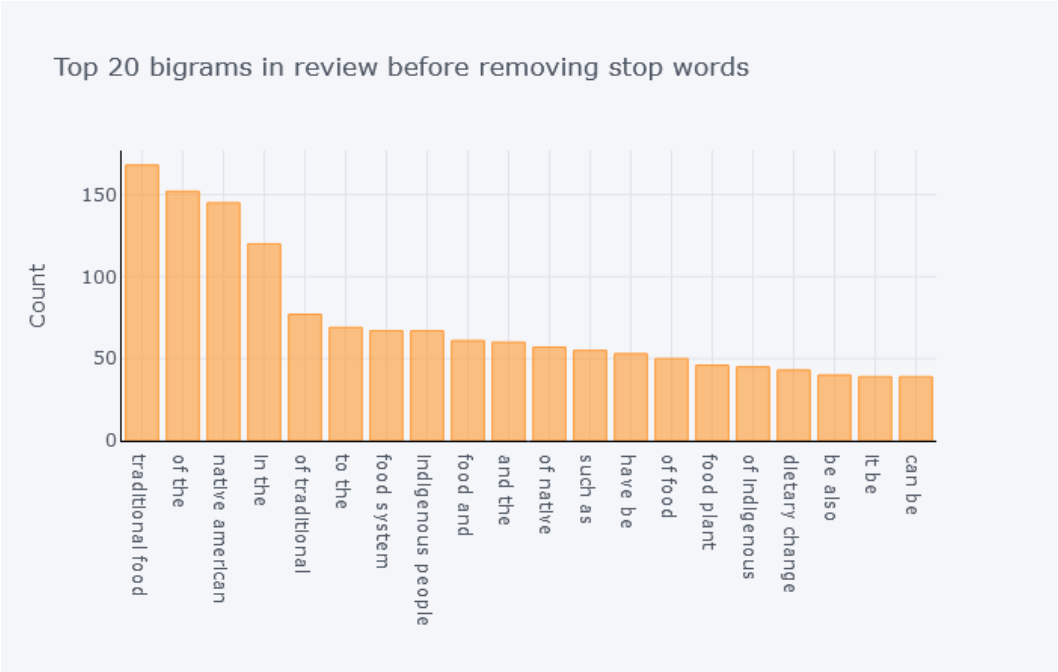

Figure 7. Top 20 bigrams from document 1 before removing the stop words.

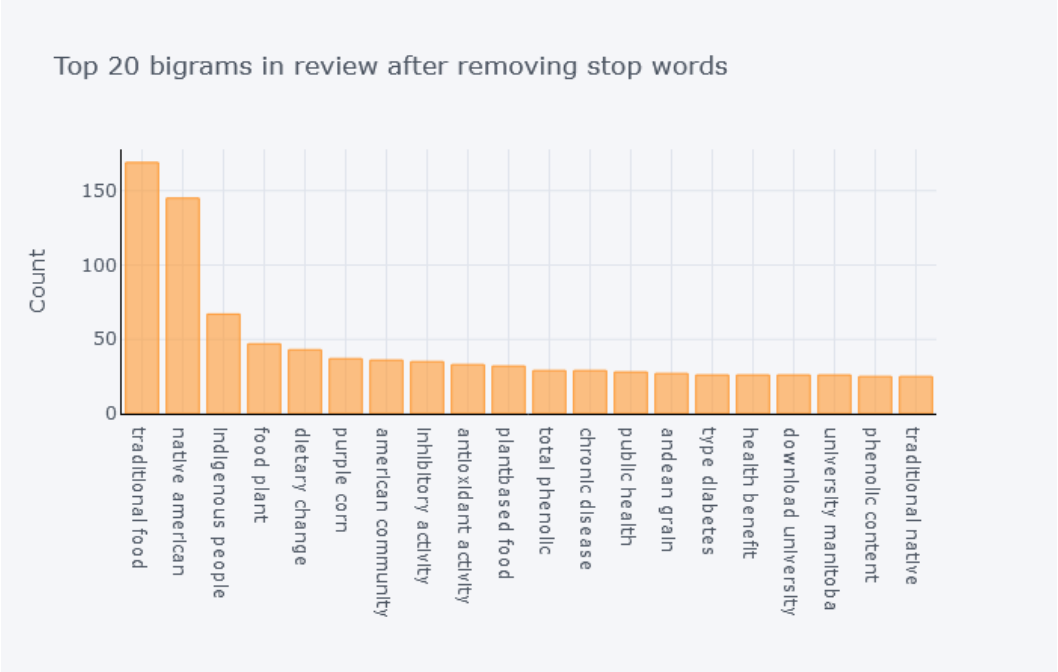

Figure 8. Top 20 bigrams from document 1 after removing the stop words.

The same analysis for bigrams also shows the importance of removing stop words from the document (Figure 7 and Figure 8).

4.2. Result Interpretation for Keyphrase Extraction

The dataset is subjected to nine algorithms to generate a list of keywords and keyphrases where the number of key phrases is set to *n,* which is equal to the number of keywords picked by the domain expert. Varying numbers of keywords/phrases were handpicked for each paper by the domain experts against which the machine-generated keywords/phrases were validated. A confusion matrix summarizing the evaluation method is generated and used to measure the Precision, Recall, and F1 scores for each algorithm under unigrams, bigrams, trigrams, and n-grams.

We use Precision, Recall, and F-score as a measure to evaluate the keywords extracted by the set of algorithms used in this study.

$$precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Table 3. Confusion matrix summarizing the evaluation method used to count True Positive, True Negative, False Positive, False Negative.

	Keywords/keyphrase picked by domain expert relevant to the topic	Keywords/keyphrase considered as irrelevant by a domain expert
Machine-generated keywords/keyphrase relevant to topic	True Positive	False Positive
Machine-generated keywords/key phrases irrelevant to a topic (Machine failed to pick the keywords generated by the domain expert)	False Negative	True Negative

F -score is the balance of harmonic average of precision and recall, therefore:

$$F_\beta = (1 + \beta^2)\frac{precision * recall}{recall + \beta^2 * precision} = \frac{(1 + \beta^2)\,TP}{(1 + \beta^2)TP + \beta^2 FN + FP}$$

where TP stands for True Positive, FP for False Positive, and FN for False Negative. β is a positive real factor (Goutte and Gaussier 2005) (Derczynski 2016). F-score is a popular metric used to evaluate NLP based systems. The highest possible value for the F-score is 1 (Derczynski 2016).

Based on the confusion matrix generated for evaluating algorithms, the keywords picked by the information extraction system are evaluated by sorting it into one of the four categories.

1. If a keyword in picked by the domain expert and the algorithm, then it falls under the category of True Positive.
2. If the domain expert picks a keyword, but the algorithm fails to classify the same phrase as a keyphrase, then it falls under False Negative.
3. Those keywords the domain expert considers not important hence does not classify as a keyphrase, yet if the algorithm considers it a keyword, it falls under False Positive.
4. Keyphrases that are considered not relevant to the topic by a domain expert and the algorithm are categorized as True Negative.

Cosine similarity is used to measure the distance between two vector space models. It is widely used in information retrieval, text classification, and other text mining techniques (Li and Han 2013), among other text similarity measurement techniques like Jaccard, Dice, Levenshtein methods. Cosine similarity is more frequently used in information mining, where angles between two vectors are measured.

Porter stemmer is used to stem the keywords from both the domain expert-picked list and the machine-generated list.

Cosine similarity is calculated as:

$$Cosine(\underline{v},\underline{w}) = \frac{\underline{v}\cdot\underline{w}}{|\underline{v}||\underline{w}|} = \frac{\sum_{i=1}^{N} v_i * w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

F-score yields a highest possible score of 1 or 100% when precision and recall are balanced (Turney 2004).

$$F - score = 2 * \frac{precision * recall}{precision + recall}$$

Based on our data, a visualization representing the evaluation method is produced (Figure 9). Precision gives the probability of the keyword being relevant to the topic. Recall gives the probability of an algorithm to pick a keyword/keyphrase as relevant to the topic (Turney 2004).

Table 4. Difference between stemmed and original keywords.

Keyword	Stemmed keyword
prevention	prevent
food	food
pepper	pepper
native	nativ
plums	plum
american	american
squash	squash
traditional	tradit
hunting	hunt
health equity	health equ
interventions	intervent
nutrition	nutrit
obesity	obes
traditional ai culture	traditional ai cultur
traditional	tradit

If the domain expert picks N number of keywords from document D, then the algorithm is also set to pick N number of keywords from the same document D. The results generated by the text mining algorithms are listed in the Table 5 where each n-gram is evaluated based on their Precision, Recall and F1-score. The data in Table 5 is categorized into 9 algorithms with each algorithm's output divided and evaluated based on 4 n-gram categories. The performance of

each algorithm is also evaluated against 5 individual documents. Precision, Recall and F-score for each category is measured, with 100% being the highest and 0% being the lowest possible scores for each evaluation metrics used. As mentioned previously, precision gives the probability of the keyword being relevant to the topic. Recall gives the probability of an algorithm to pick a keyword/keyphrase as relevant to the topic (Turney 2004), but (Goutte and Gaussier 2005) mentions a scenario where a small number of True Positives (In this research relevant keywords) and a larger True Negative, False Positive and False Negative will result in a high fluctuation in Recall value, which is also observed in this research. One such scenario would be in document 2 where the TextRank algorithm generates a Recall value of 100% while precision is just 4% and F-score is 8%, due to very few True Positives. Under the above-mentioned scenarios, the very particular result is discarded or set to 0. Since F-score is widely used to evaluate NLP-based systems, F-score was used to pick out the best performing algorithm in the upcoming sections. The results shown in Table 5 and Figure 15 is based on the analysis of 9 algorithms.

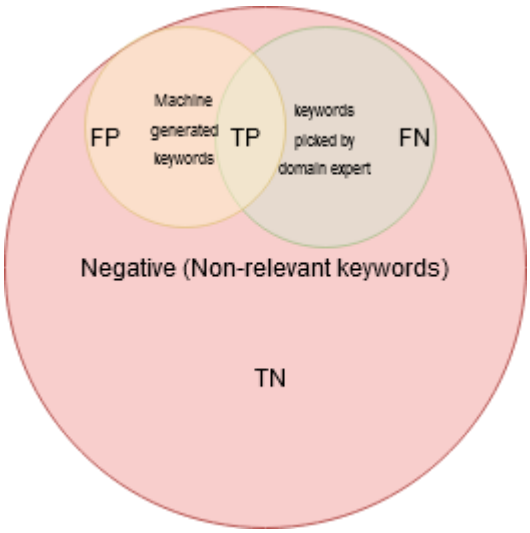

Figure 9. Visualization of evaluation metrics used to measure the performance of keyword extraction algorithm.

Table 5. Results of automatic keyword extraction using statistical, graph-based, and machine learning based algorithms, D1 represents document 1, similarly, D2, D3, D4 and D5 represent Document2, Document3, Document4, Document5. P denotes Precision, R for Recall, F for F-score.

		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
TFIDF	One gram	35	14	20	35	16	22	22	10	14	19	14	16	15	12	14
	Bigram	16	23	19	23	24	24	16	21	18	15	15	15	12	23	16
	Trigram	4	6	5	10	12	11	11	22	14	0	0	0	0	0	0
	4-gram	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
YAKE	One gram	50	20	29	37	17	23	19	8	11	16	12	14	20	22	21
	Bigram	29	49	36	29	43	35	18	39	25	15	19	17	12	30	18
	Trigram	15	17	14	21	16	18	16	20	18	0	0	0	0	0	0
	4-gram	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
MultipartiteRank	One gram	32	16	21	29	14	19	22	10	14	16	19	18	35	25	29
	Bigram	20	26	23	28	30	29	16	21	18	25	21	23	29	54	38

Table 5. Results of automatic keyword extraction using statistical, graph-based, and machine learning based algorithms, D1 represents document 1, similarly, D2, D3, D4 and D5 represent Document2, Document3, Document4, Document5. P denotes Precision, R for Recall, F for F-score (continued).

	Trigram	4	5	4	17	22	19	11	22	14	0	0	0	0	0	0
	4-gram	0	0	0	32	38	34	0	0	0	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
PositionRank	One gram	9	50	15	10	50	16	4	17	6	3	50	6	0	0	0
	Bigram	41	36	38	41	38	40	26	19	22	20	17	19	4	11	6
	Trigram	46	18	26	53	23	32	42	19	26	17	5	8	0	0	0
	4-gram	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
SingleRank	One gram	6	0	11	6	75	11	4	25	6	0	0	0	0	0	0
	Bigram	28	45	34	29	50	37	18	21	25	21	5	0	1	4	0
	Trigram	35	18	24	39	27	32	37	32	34	0	0	0	0	0	0
	4-gram	9	3	4	42	13	2	5	16	24	0	0	0	0	0	0

Table 5. Results of automatic keyword extraction using statistical, graph-based, and machine learning based algorithms, D1 represents document 1, similarly, D2, D3, D4 and D5 represent Document2, Document3, Document4, Document5. P denotes Precision, R for Recall, F for F-score (continued).

		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
TextRank	One gram	9	100	16	4	100	8	0	0	0	0	0	0	0	0	0
	Bigram	32	51	39	24	49	32	16	22	18	5	50	9	0	0	0
	Trigram	35	20	25	41	28	33	32	26	29	0	0	0	0	0	0
	4-gram	9	3	4	42	13	20	50	16	24	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
TopicalPageRank	One gram	6	67	11	6	75	11	4	33	7	0	0	0	0	0	0
	Bigram	36	45	40	34	47	39	24	28	26	10	25	14	4	33	7
	Trigram	42	21	28	50	29	37	37	26	30	8	4	6	0	0	0
	4-gram	9	4	5	42	15	22	50	17	25	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
KEA	One gram	18	11	13	14	7	9	7	5	6	13	15	14	5	5	5
	Bigram	12	17	14	7	9	8	5	7	6	5	5	5	12	27	17

Table 5. Results of automatic keyword extraction using statistical, graph-based, and machine learning based algorithms, D1 represents document 1, similarly, D2, D3, D4 and D5 represent Document2, Document3, Document4, Document5. P denotes Precision, R for Recall, F for F-score (continued).

KEA	Trigram	0	0	0	0	0	0	5	4	4	0	0	0	0	0	0
	4-gram	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		D1			D2			D3			D4			D5		
		P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
WINGUS	One gram	32	19	24	18	10	13	7	4	6	10	17	12	5	4	5
	Bigram	19	24	21	25	26	26	16	21	18	20	17	18	12	27	17
	Trigram	8	5	6	27	20	23	32	23	27	0	0	0	0	0	0
	4-gram	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The experiments indicate that unsupervised methods, specifically Graph-based methods, outperform other methods for keyphrase extraction (bigrams and trigrams). Picking the SingleRank output for document 5 (Table 5) the rate of Recall is 1.0 which raised some questions. Further analysis showed that the True Positive is 1, False Positive is 23, True Negative is 8625, and False Negative is 0 for that particular result. In this specific instance, the chances of getting a False Positive is very high due to higher negative classes (non-keywords) while there is a lower chance of getting False Negatives. This gives a false recall value, of 1.0, which in this case is discarded. Compared to all other algorithms, TopicalPageRank performed the best when it comes to keyword extraction from document 1 with an F1 score of 40%, precision of 36% and recall of 45% followed by TextRank with an F1 score of 39%, precision of 32% and recall of

51% for bigrams. PositionRank is the 3rd best algorithm with an F1 score of 38%, with precision and accuracy of 41% and 36%, respectively. The data analysis shows that graph-based algorithms outperformed all other methods, including statistical and machine learning, when it comes to bigram extraction (key phrases). Among statistical models, the YAKE performed the best with an F1 score of 36%, recall of 49%, and precision of 29% for Document1.

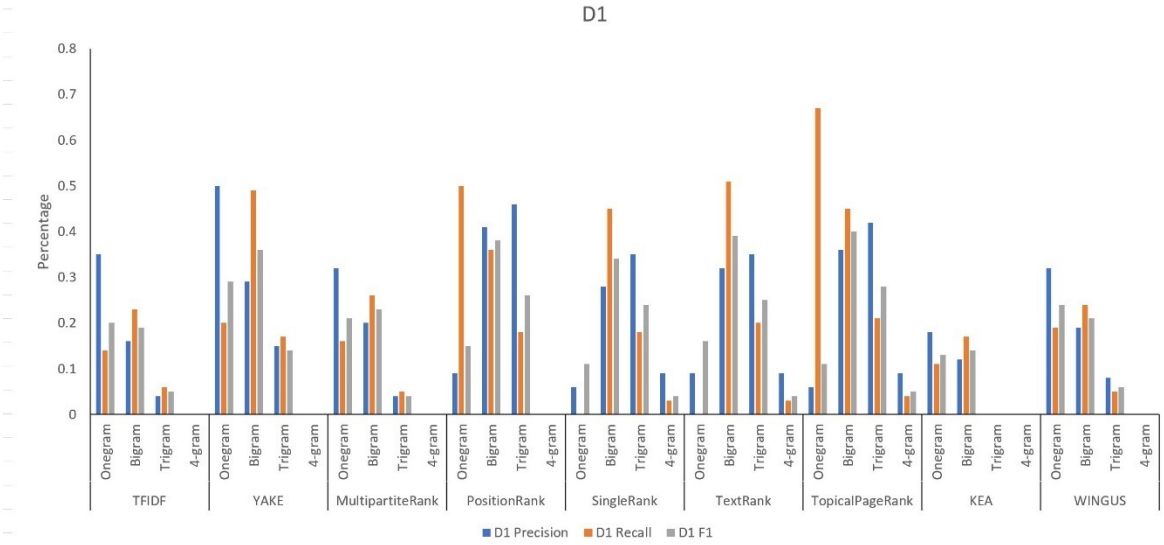

Figure 10. Evaluation of algorithms for Document 1, based on Precision, Recall and F1-score for various n-grams.

Document 2 has about 5422 words with 1227 unique words which also has the highest number of keywords/phrases picked by domain experts (284). The results for unigrams picked by graph-based methods like PositionRank, SingleRank, TextRank, and TopicalPageRank show higher recall rates due to imbalanced datasets. The best algorithm for document 2 would be PositionRank with 40% F1 score, 41% precision and 38% recall for trigrams followed by 39% F1-score for TopicalPageRank with 34%, 47% precision and recall respectively. It is also noticeable that these two algorithms performed best when picking trigrams and four-grams. Statistical models like YAKE and TFIDF performed well at picking unigrams. YAKE gives 29%

precision with 43% recall and 35% F1 score for bigrams followed by TFIDF with 23% recall, with 24% recall and 24% F1.

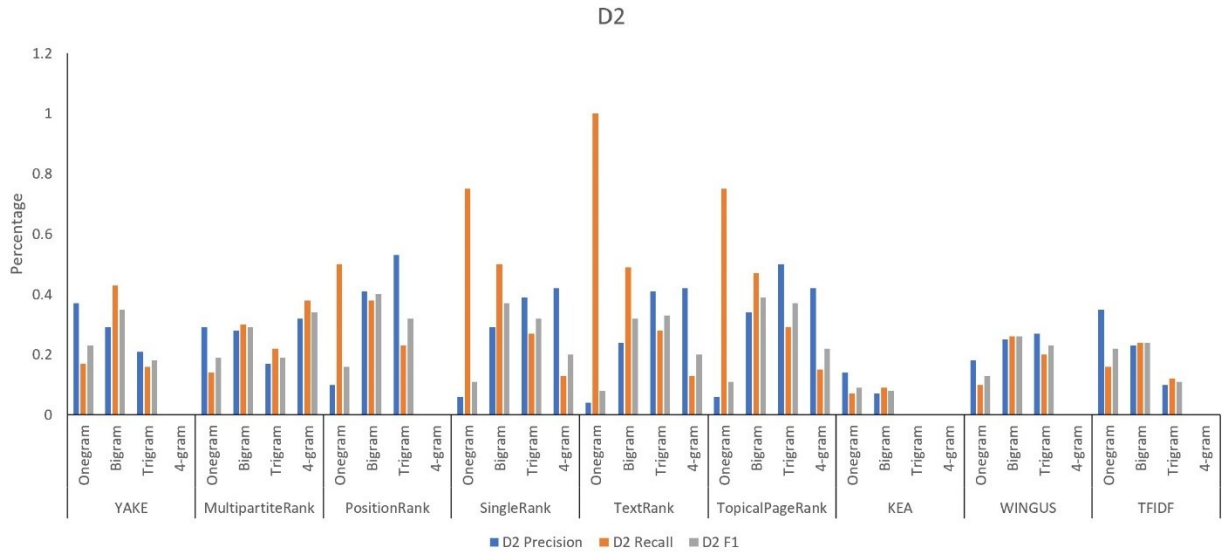

Figure 11. Evaluation of algorithms for Document 2, based on Precision, Recall and F1-score for various n-grams where PositionRank shows best performance.

Document 3 has about 3615 words with 1007 unique words. Based on the results it shows that SingleRank performed the best with 34% F1 score, 37% precision and 32% recall for trigrams followed by 24% F1 score, 50% precision and 16% recall for four-grams (Figure 12). Other Graph-based methods like TopicalPageRank, TextRank and PositionRank also performed well with 30%, 29% and 26% F1-score values. TFIDF and MultipartiteRank with a precision of 22% performed best when it comes to picking unigrams from this document. Based on the analysis so far YAKE ranks second when it comes to picking the right unigrams. WINGUS, a supervised keyword extraction algorithm exhibited good performance in picking trigrams with an F1-score of 27%.

Document 4 with about 1495 unique words, among which the domain expert picked 66 keywords/phrases. Compared to the first 3 documents, documents 4 and 5 performed poorly

overall, when it comes to automatic keyword extraction (Figure 13). The highest score was achieved by MultipartiteRank with an F1 score of 23%, with a precision of 25% and a recall of 21%. PositionRank and WINGUS stand next in line when it comes to performance with F1 scores of 19% and 18% respectively. Once again bigrams give the best result compared to other n-grams.

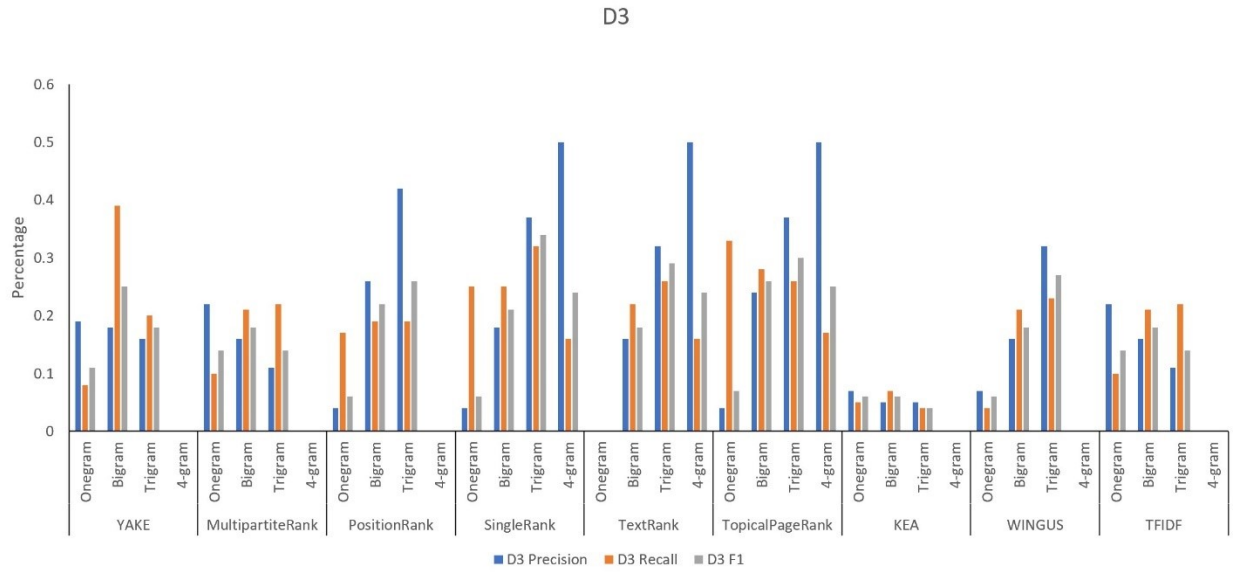

Figure 12. Evaluation of algorithms for Document 3, based on Precision, Recall and F1-score for various n-grams where SingleRank shows the best performance.

Document 5 with 1989 unique words performed the least when it came to automatic keyword extraction. This document being one with just 45 keywords picked by the domain expert while having the highest number of unique words. The above-mentioned scenario results in a very low True Positive to True Negative ratio resulting in lower precision, recall, and F1 score. Some of the algorithms like TextRank, which performed the best with all other documents, did not generate any unigrams, bigrams, trigrams, or four-grams with document 5 (Figure 14). In addition to that, SingleRank also developed a false recall due to an imbalance in the dataset. Ranking the ranking algorithms, MultipartiteRank has the best F1 score with 38%,

precision of 54%, and recall of 29% for bigrams, while the second-best keyword list was generated for unigram by the same algorithm. YAKE follows MultipartiteRank with an F1 score of 21%, which is the best statistical model.

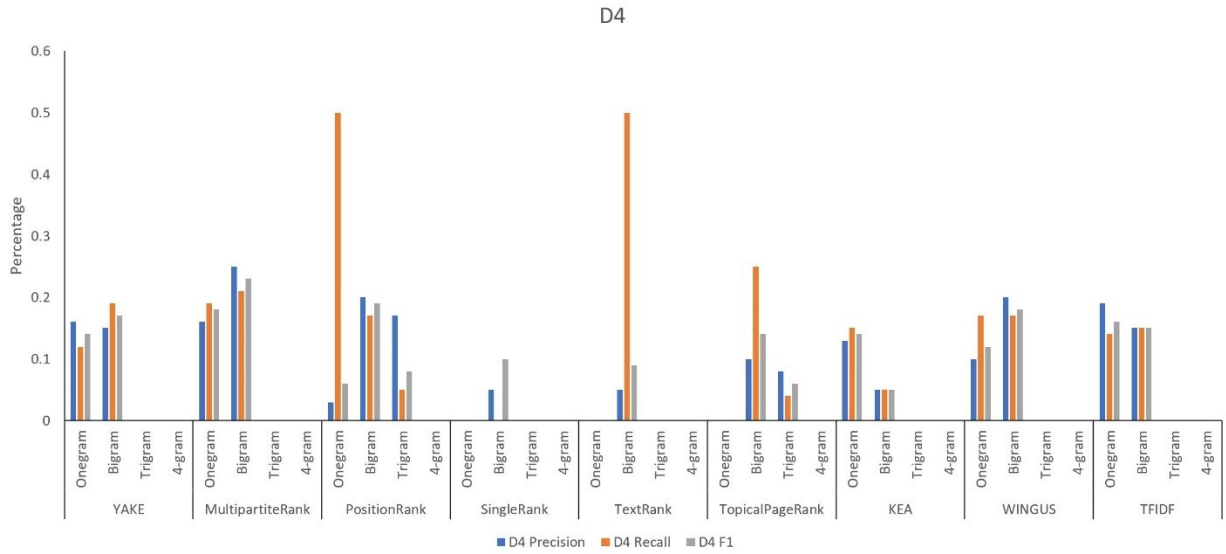

Figure 13. Evaluation of algorithms for Document 4, based on Precision, Recall and F1-score for various n-grams where MultipartiteRank shows the best performance.

The overall results show that statistical methods like TFIDF and YAKE perform best picking up unigrams whose performance on bigrams are also reasonable. Bigrams had the highest scores when it comes to keywords extraction compared to all other n-grams. When it comes to bigrams, graph-based keyword/phrase extraction methods performed the best.

Based on the data represented in Table 6, MultipartiteRank, a graph-based data mining algorithm, performed the best in documents with a lower number of keywords. At the same time, TopicalPageRank, PositionRank, and SingleRank showed good performance on documents with a higher number of domain-specific keywords. It is also observed that four out of the five best results were generated with bigrams followed by unigrams and trigrams.

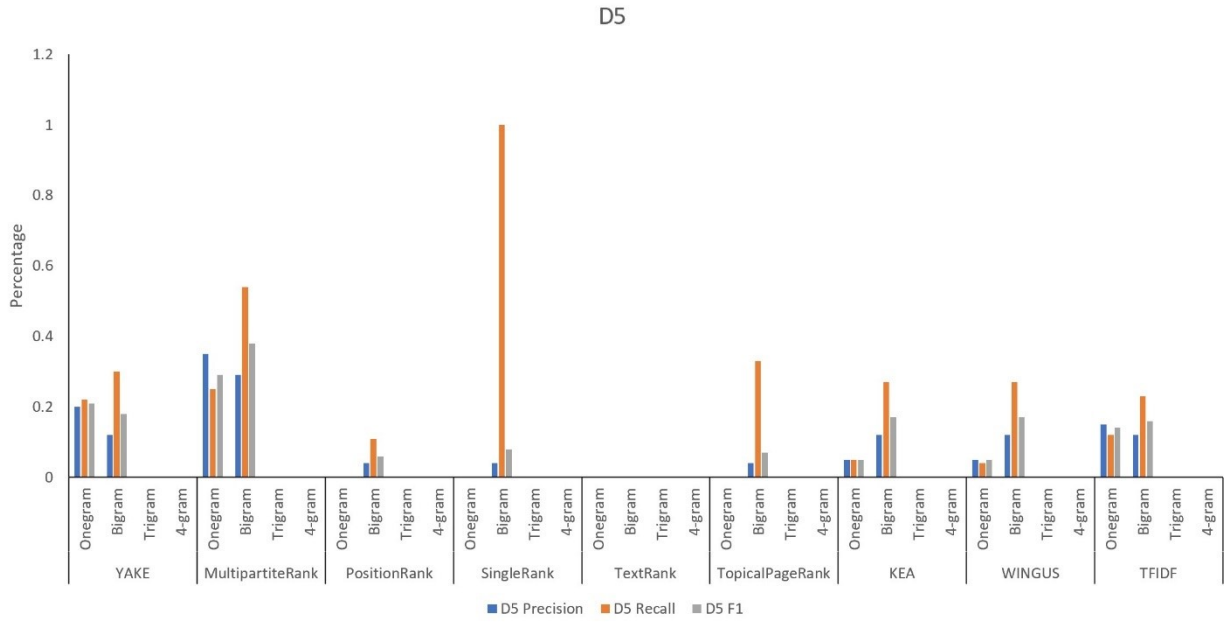

Figure 14. Evaluation of algorithms for Document 5, based on Precision, Recall and F1-score for various n-grams where MultipartiteRank shows the best performance for bigrams.

Table 6. Best algorithm performed based on F1-score at picking keywords based on document.

	Document				
	D1	D2	D3	D4	D5
Performance-based on F1 score	TopicalPageRank (bigrams)	Position Rank (bigrams)	SingleRank (trigrams)	MultipartiteRank (bigrams)	MultipartiteRank (bigrams)

Each algorithm picks keywords and phrases from 4 classes of n-grams, unigram, bigram, trigram, and four grams. All nine algorithms generate keywords falling into the above-mentioned categories of n-gram generating a result of 9*4 = 36 sections. While certain algorithms are good at picking a specific n-gram (YAKE is good at generating unigrams) there is a correlation of the algorithm generated keywords with the document size, number of keywords picked by the domain expert and total number of unique keywords present in the document (Figure 15).

To conclude, according to our analysis TopicalPageRank and PositionRank are the algorithms with the highest F1 score and hence considered best. Evaluating the tested algorithms, Graph-based text mining algorithms outperformed both statistical and Machine Learning based algorithms. It is also worth mentioning that YAKE had the best results in statistical models and WINGUS with an F1 score of 27 was observed to perform well compared to KEA for our dataset. In conclusion, considering the application of these algorithms for further research and development, it is more suitable to pick the set of best algorithms. These therefore are, YAKE, MultipartiteRank for unigrams, PositionRank, TopicalPageRank for bigrams or MultipartiteRank for larger documents with fewer keywords.

Table 7. Table of best performing algorithms based on F1 score for different n-gram.

	Unigrams	bigrams	trigrams	four-grams
Best Performed Algorithm (F1 score)	YAKE, MultipartiteRank	PositionRank, TopicalPageRank	TopicalPageRank	MultipartiteRank

4.3. Result Interpretation for Topic Modeling

The outcome of different topic modeling methods is a set of topics with each topic having its list of top *n* key phrases. Once the topics are generated, each paper is categorized and assigned to the most suited topic. After the results are generated, they are evaluated by a domain expert to check if the grouping of papers is logical and makes sense. The domain expert closely examines the result by looking at the topic and Phi values along with document and its calculated weight. Based on the domain expert's evaluation the best algorithm is rated and selected as the final best.

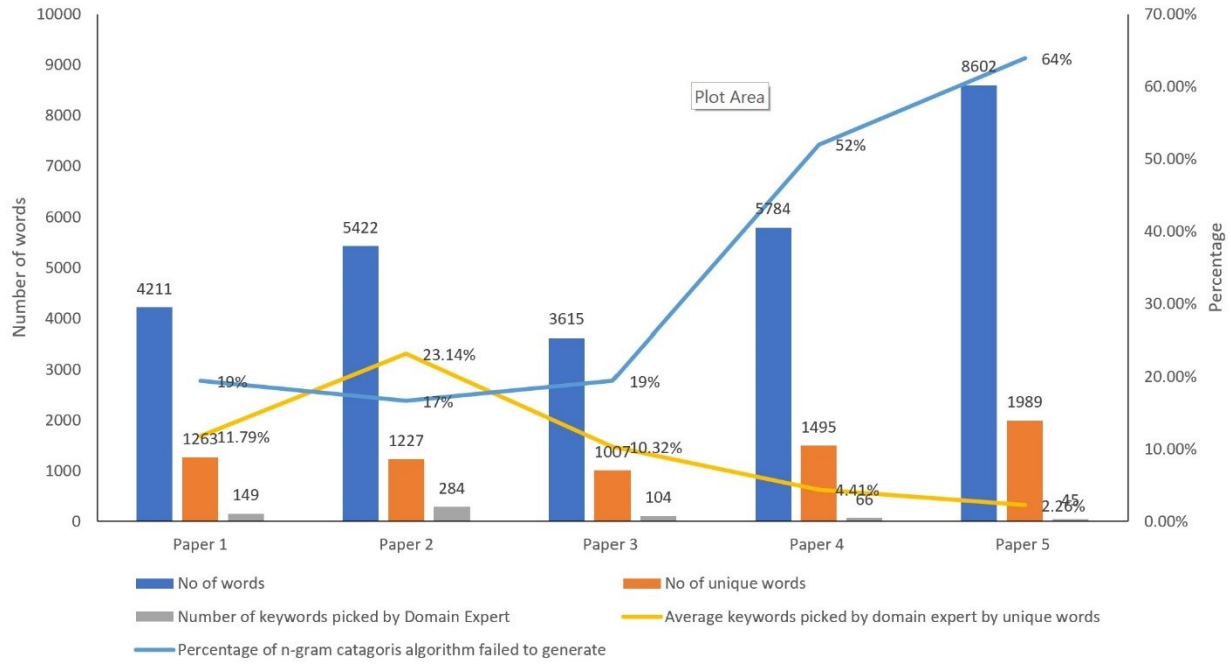

Figure 15. Depicts the size of the document, and more importantly the number of keywords picked by the domain expert being directly proportional to the quality of the machine-generated output.

4.3.1. Latent Dirichlet Allocation

The LDA algorithm is run with a bag-of-words and keyphrase list, where the algorithm is set to generate three topics in total. Once the topic list is generated, a document-topic weight matrix is generated where each 5 individual papers are allocated into its respective topic.

When a list of keywords, BOW or TF IDF frequency passed as a parameter, the algorithm returns:

1. Document classification (along with the probability of that document belonging to that topic)
2. Each word categorized into respective topic
3. Phi value of each word (probability of a keyword belonging to a particular topic)

Based on the table described above, the documents are classified into their respective topics based on the phi values of each keyword (weights). Phi value is the probability of a

keyword belonging to a particular topic. For example, the keyword '*diet'* has a phi value of 0.087 (table 8), which indicates that given the word diet appearing across all the topics, the probability of it belonging to topic 2 is 0.087, which is higher than it belonging in topic 3 with 0.014 probability.

Evaluation of Indigenous Grains from the Peruvian Andean Region for Antidiabetes and Antihypertension Potential Using *In Vitro* Methods

Lena Galvez Ranilla,[1] Emmanouil Apostolidis,[2] Maria Ines Genovese,[1] Franco Maria Lajolo,[1] and Kalidas Shetty[2]

[1]Laboratório de Química, Bioquímica e Biologia Molecular de Alimentos, Departamento de Alimentos e Nutrição Experimental, Faculdade de Ciências Farmacêuticas, Universidade de São Paulo, São Paulo, Brazil; and [2]Department of Food Science, Chenoweth Laboratory, University of Massachusetts, Amherst, Massachusetts, USA

ABSTRACT The health-relevant functionality of 10 thermally processed Peruvian Andean grains (five cereals, three pseudocereals, and two legumes) was evaluated for potential type 2 diabetes-relevant antihyperglycemia and antihypertension activity using *in vitro* enzyme assays. Inhibition of enzymes relevant for managing early stages of type 2 diabetes such as hyperglycemia-relevant α-glucosidase and α-amylase and hypertension-relevant angiotensin I-converting enzyme (ACE) were assayed along with the total phenolic content, phenolic profiles, and antioxidant activity based on the 1,1-diphenyl-2-picrylhydrazyl radical assay. Purple corn (*Zea mays* L.) (cereal) exhibited high free radical scavenging-linked antioxidant activity (77%) and had the highest total phenolic content (8 ± 1 mg of gallic acid equivalents/g of sample weight) and α-glucosidase inhibitory activity (51% at 5 mg of sample weight). The major phenolic compound in this cereal was protocatechuic acid (287 ± 15 μg/g of sample weight). Pseudocereals such as Quinoa (*Chenopodium quinoa* Willd) and Kañiwa (*Chenopodium pallidicaule* Aellen) were rich in quercetin derivatives (1,131 ± 56 and 943 ± 35 μg [expressed as quercetin aglycone]/g of sample weight, respectively) and had the highest antioxidant activity (86% and 75%, respectively). Andean legumes (*Lupinus mutabilis* cultivars SLP-1 and H-6) inhibited significantly the hypertension-relevant ACE (52% at 5 mg of sample weight). No α-amylase inhibitory activity was found in any of the evaluated Andean grains. This *in vitro* study indicates the potential of combination of Andean whole grain cereals, pseudocereals, and legumes to develop effective dietary strategies for managing type 2 diabetes and associated hypertension and provides the rationale for animal and clinical studies.

KEY WORDS: • *α-amylase inhibitory activity* • *Andean grains* • *angiotensin I-converting enzyme* • *antioxidant activity* • *α-glucosidase inhibitory activity* • *hypertension* • *phenolic phytochemicals* • *type 2 diabetes*

Figure 16. Visualization of automatic keyphrase extraction with Multipartite rank algorithm, MeSH, and Plant term database on D4 (Ranilla et al. 2009). The blue highlight indicates the keywords picked by Multipartite Rank, purple for MeSH terms (database assisted), green for common names of plant species, orange for scientific names.

The results on Table 9 show that Topic 3 occupies about 99% of the total topic in document D1 and D4, Topic 2 occupies about 99% of the total topic in document D2 and D3, and Topic 1 occupies about 99% in document D5 which is based on the mathematical principle that summing up all weights on each row is 100. The results obtained here shows a clear

indication that documents are addressing different variants of the main topic of interest Figure 18.

RESULTS

Traditional Dietary Components:

Elders were asked what foods they considered to be traditional Native American foods. Foods primarily identified as being staples in the traditional Native American diet by the Elders included: prairie turnips, fruits (chokecherries, June berries, plums, blueberries, cranberries, strawberries, buffalo berries, gooseberries), potatoes, squash, dried meats (venison, buffalo, jack rabbit, pheasant, and prairie chicken), corn, teas (spearmint, peppermint, bergamot), and wild rice (see Table 2). All of the identified plants were perennial, primarily grew and were harvested in summer and early fall, and were dried for use over the winter. These foods were reported to have been eaten for generations. Additional information on harvesting, preparation, storage, and frequency of consumption was collected.

ing in many types of dishes. Ground prairie turnips were used as a thickening agent. Breads were made out of corn, turnips, and later, wheat. Meat, corn, crushed berries, sliced turnips, and sliced squash were all often dried on roof tops. Children attended to the drying foods on roof tops by turning the foods over and defending the foods from animals. Meats were sliced thinly to avoid spoilage. Berries were also made into syrups and jams. Wasna was made from dried meat (traditionally buffalo), crushed dried berries (often chokecherries) and rendered animal fat. Corn balls were made with crushed dried corn, crushed dried berries, and rendered animal fat. Rice was used in many ways as it could be popped like popcorn or it could be cooked and later used like oatmeal. Rice was also added to breads.

Preparation:

Elders were asked how common traditional Native American foods were often prepared. Prairie turnips, carrots, squash, corn and dried meat were reportedly used in soups. Tipsila or washtunkala soup consisted primarily of dried meat, turnips and corn. Herbs, spices and salt were not used in traditional dishes; however, wild onions were used frequently for flavoring in many types of dishes. Ground prairie turnips were used as a thickening agent. Breads were made out of corn, turnips, and later, wheat. Meat, corn, crushed berries, sliced turnips, and sliced squash were all often dried on roof tops. Children attended to the drying foods on roof tops by turning the foods over and defending the foods from animals. Meats were sliced thinly to avoid spoilage. Berries were also made into syrups and jams. Wasna was made from dried meat (traditionally buffalo), crushed dried berries (often chokecherries) and rendered animal fat. Corn balls were made with crushed dried corn, crushed dried berries, and rendered animal fat. Rice was used in many ways as it could be popped like popcorn or it could be cooked and later used like oatmeal. Rice was also added to breads.

Frequency of consumption:

Elders were asked how often different types of traditional foods were consumed. Soup and tea was reported to be consumed almost daily. Teas mentioned included: a variety of mint (including peppermint), sage, yarrow, bergamot, anise, blueberry bark, bitterroot, licorice root, and chamomile. Teas were often flavored with berries. Wasna was often consumed daily and throughout winter.

Figure 17. Visualization of automatic keyphrase extraction with TopicalPageRank algorithm, MeSH, and Plant term database on Document D3 (Colby, McDonald, and Adkison 2012). The blue highlight indicates the keywords picked by TopicalPageRank, purple for MeSH terms (database assisted) automatic keyword extraction, green for common names of plant species, orange for fruit names, red is manually highlighted where the keyphrases were not picked.

Table 8. List of topics, weights, and keyphrase generated by Latent Dirichlet Allocation.

Topic 1		Topic 2		Topic 3	
Weights	words	Weights	words	Weights	words
0.037402585	nutrition	0.08715588	diet	0.051917884	native american
0.035565984	health	0.08613036	traditional food	0.028777305	native american communities
0.022685003	social determinants	0.0602812	indigenous peoples	0.020768283	traditional foods
0.019008601	poverty	0.053051613	nutrition	0.01898703	traditional food plants
0.019006334	breastfeeding	0.940646706	dietary change	0.01898703	indigenous communities
0.017166993	food programs	0.030312767	purple corn	0.016316961	indigenous
0.017166993	tribes	0.02927937	antioxidant activity	0.015429034	health benefits
0.013495386	diabetes	0.028245976	andean grains	0.015425004	health
0.013489664	health disparities	0.026179193	culture	0.014531401	diet
0.013488328	tribal	0.025147676	type 2 diabetes	0.010976822	native ecosystem
0.013488328	social determinants of health	0.024111528	health	0.010976822	native American tribes
0.013488328	healthy food	0.02308089	quinoa	0.010971861	nutrition
0.011657755	obesity	0.01895176	hypertension	0.010087661	squash
0.011648989	access	0.014811847	pseudocereals	0.010086779	traditional native american foods

Table 8. List of topics, weights, and keyphrase generated by Latent Dirichlet Allocation (Continued).

Topic 1		Topic 2		Topic 3	
Weights	words	Weights	words	Weights	words
0.011648989	heart disease	0.013778452	food	0.009197733	diversity
0.011648989	traditional	0.012745065	disease	0.009197211	traditional knowledge
0.011648989	nutritional health	0.011714441	corn	0.009196777	traditional plant based foods
0.011648989	american indian	0.010678992	hunting	0.008306754	food plants
0.009809655	death	0.010678276	dietary structure	0.008306752	native
0.009809655	access to traditional foods	0.010678266	hyperglycemia	0.008306747	reservations

4.3.2. Non-Negative Matrix Factorization

NMF algorithm is run with the parameters, number of topics = 3. The Python Sklearn package is used to carry out the document classification here. A document vector generated by the TFIDF model is used as a feature vector, which is transformed, normalized, and passed to the algorithm.

The outputs of the algorithm are listed in Table 10. NMF explains document features by generating product two matrices, W and H. Table 11 shows Topic 1 is a feature that is highly related to the frequency of words frequently appearing in D3 and D5 with a highest score of 0.54, similarly Topic 2 in D2, while Topic 3 in D1 and D4. The results in Table 11 show that Topic 1 occupies about 54% of the total topic in document D3, Topic 2 occupies about 48% of

the total topic in document D2, and Topic 3 occupies about 39% and 10% of the topic in document D1 and D4 respectively. A visual representation of inter-topic distance is represented in Figure 20. While key phrases like ‘native,’‘American’, and ‘indigenous’ are standard terms that identify the main topic, the rest of the keywords better identify the sub-topics addressed by the scholarly articles chosen for this study.

Table 9. Document classification based on the list of keywords generated topic by Latent Dirichlet Allocation.

Document	Topic 1 weight	Topic 2 weight	Topic 3 weight
D1	0.0017	0.0017	0.99
D2	0.0011	0.99	0.0011
D3	0.0006	0.99	0.0006
D4	0.0004	0.0004	0.99
D5	0.99	0.0009	0.0009

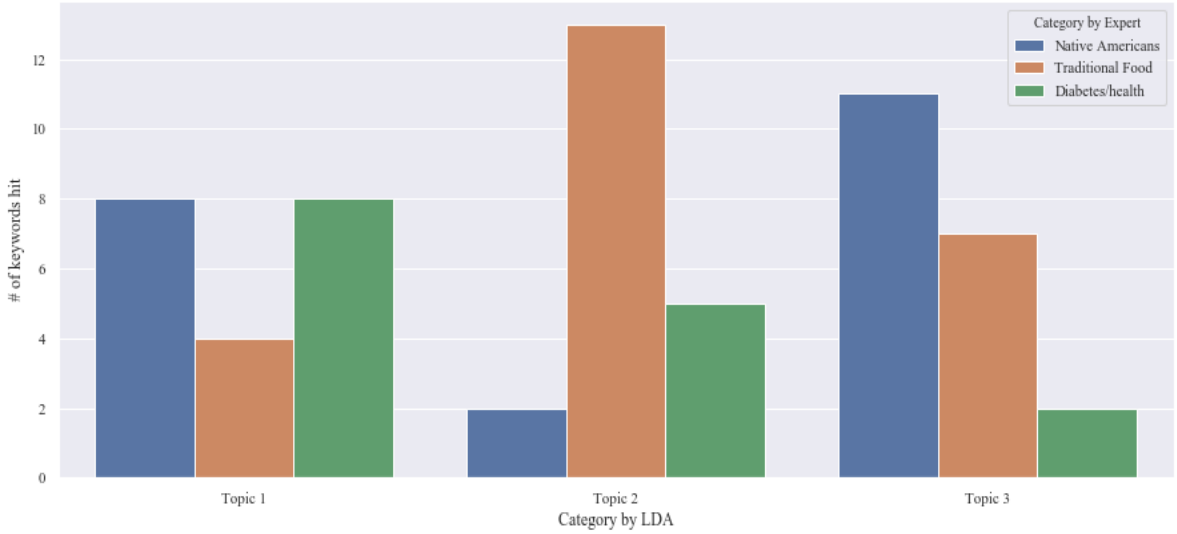

Figure 18. Distribution 3 different domain-related topics (Native Americans, Traditional food, and Diabetes/health) addressed in 3 topics generated by LDA.

Table 10. List of topics, weights, and keyphrase generated by Non-zero Matrix Factorization.

Topic 1	Topic 2	Topic 3
diet	purple corn	native american
indigenous peoples	antioxidant activity	traditional native american foods
traditional food	andean grains	tradition
dietary change	type 2 diabetes	traditional food
nutrition	quinoa	mint
culture	pseudocereals	prairie turnip
health	hyperglycemia	native american foods
disease	food	dried meat
dietary structure	hypertension	native american researcher
environmental	anthocyanins	turnips
ecology	antioxidant	obesity and diabetes
hunting	anti-hypertension	bergamot
traditional food systems of indigenous peoples	glucose	corn
corn	phenolic profiles	traditional native american diet
fishing	tarwi	peppermint
food selection	health relevant	traditional dietary practices
meal	cultivated	chokecherries
traditional cultural	glycemic index	native american communities
traditional knowledge	diversity	wasna

Comparing the document classification generated by LDA and NMF, except for document 3 all the documents were classified similarly by both the algorithms. Although when it

comes to topic list and document classification, the accuracy of the list of words and the topic classification must both make sense.

Table 11. Document classification based on the list of keywords generated by Non-zero Matrix Factorization, topics categorized with coherent score described in section 4.2.1.

Document	Topic 1 weight	Topic 2 weight	Topic 3 weight
D1	0.0	0.0	0.39
D2	0.0	0.48	0.0
D3	0.54	0.0	0.0
D4	0.03	0.00	0.10
D5	0.02	0.00	0.00

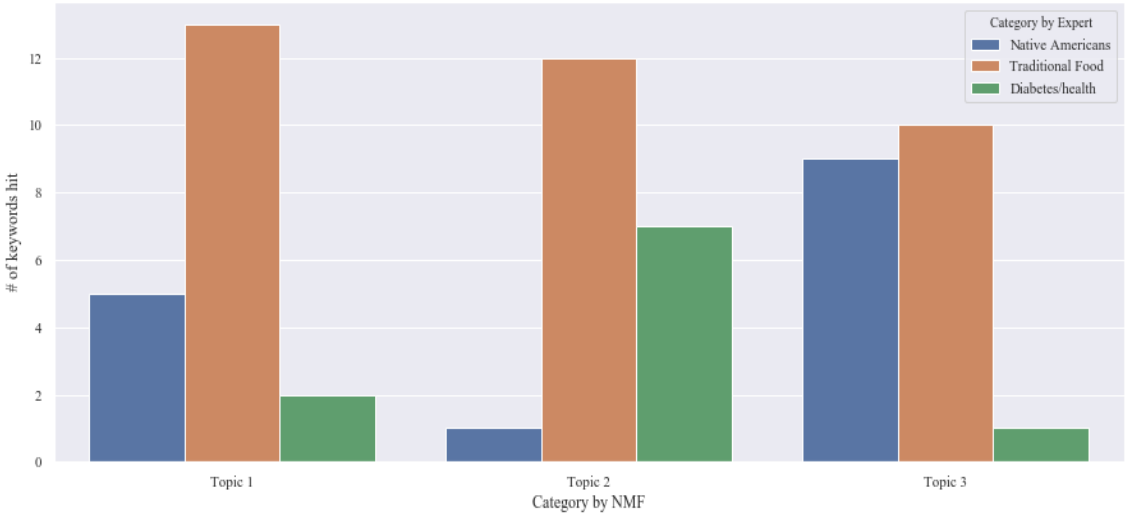

Figure 19. Distribution 3 different domain-related topics (Native Americans, Traditional food and Diabetes/health) addressed in 3 topics generated by NMF.

A comparison by the domain expert from food science showed that the topic list generated by Non-negative Matrix Factorization generated a better set of topics. It also exhibits a higher accuracy and reliability of classification compared to Latent Dirichlet Assignment. Although LDA is known as a state-of-the-art algorithm for topic modeling, NMF performed better in our scenario.

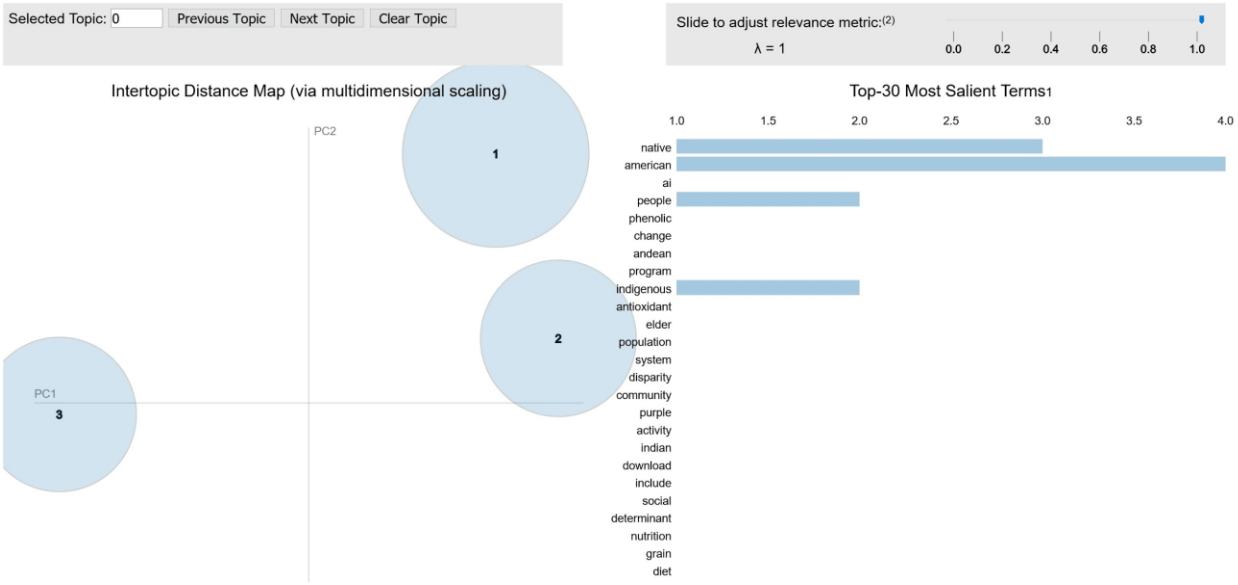

Figure 20. Visualization of NMF overall term frequency across documents using matrix H and checking which topic has the highest score for each document. The blue bar chart represents the overall term frequency.

Table 12. Best algorithms picked by the study.

Best keyword extraction algorithm	Best topic modeling algorithm
1. TopicalPageRank 2. PositionRank	1. Non-zero Matrix Factorization

5. CONCLUSION AND FUTURE SCOPE

This thesis implemented various statistical, graph-based and machine learning based models for keyword extraction and topic modeling. Although it gives a good result for real-world application, the scope and potential of this work is much higher and therefore more methods need to be researched to discover better algorithms.

Building an exploratory literature review toolkit always starts with identifying the right documents, hence one good set of useful keyword/phrase extraction algorithms is the first and most crucial step. Once the documents are extracted, classifying them would be the next major step. Classification is essential for indexing, search and retrieval of the gathered information from documents.

5.1. Goal 1 - Keyword/Phrase Extraction

Based on this research and literature review, Graph-based algorithms perform better in comparison to Machine Learning based keyword extraction algorithms. While machine learning gives an acceptable result, the scope of Deep Learning on extracting important concepts and terms from food science scholarly articles could be substantial. One of the key advantages of Deep Learning is its self-learning ability and its high performance. While deep learning offers a potential improvement to the current research in this thesis, the lack of data is a bottleneck to our research. Some of the existing deep learning frameworks for keyword extraction include bidirectional Long Short-Memory (LSTM), Doc2vec, Word2vec and more.

The future goal from this study is to gather more scholarly articles using the existing methods from our metabolically relevant food research. Development of a good training dataset and implementation of various deep learning architectures and algorithms for the improvement of results is also a goal.

5.2. Goal 2 - Scraping Data, Indexing and Building a Search Engine

The goal of this research is to mine more data and present useful information from the gathered data to researchers. Information can be retrieved from public databases like Pubmed, Nature and more. These databases must be queried based on the set of keywords picked by the domain expert and generated by the algorithms. A set of papers are returned which is then matched by the best algorithm picked by this research. The retrieved documents must be classified into sections and indexed for a search engine.

5.3. Goal 3 - Toolkit Development

The use of computer programming to automate the process of gathering and retrieving information from scholarly articles from the World Wide Web (WWW) requires expertise and skills that have a sharp learning curve. A Graphical User Interface (GUI) needs to be developed where the domain expert can enter a query related to the domain of interest and the search results returns information from the indexed content.

REFERENCES

Adnan, Kiran, and Rehan Akbar. 2019. "Limitations of information extraction methods and techniques for heterogeneous unstructured big data." *International Journal of Engineering Business Management* 11:1-23. 10.1177/1847979019890771.

Albalawi, Rania, Tet H. Yeap, and Morad Benyoucef. 2020. "Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis. Front." *Frontiers in Artificial Intelligence* 3 (7): 42. 10.3389/frai.2020.00042.

Anupriya, P., and S. Karpagavalli. 2015. "LDA based topic modeling of journal abstracts." *2015 International Conference on Advanced Computing and Communication Systems*, 1-5.

Asmussen, Claus Boye, and Charles Møller. 2019. "Smart literature review: a practical topic modelling approach to exploratory literature." *Journal of Big Data* 6, no. 93 (10). https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0255-7.

Azcarraga, Arnulfo, Michael D. Liu, and Rudy Setiono. 2012. "Keyword extraction using backpropagation neural networks and rule extraction." *The 2012 international joint conference on neural networks (IJCNN)*, (7), 1-7. 10.1109/IJCNN.2012.6252618.

Bird, Steven, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* N.p.: O'Reilly Media, Inc.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. "Latent dirichlet allocation." *Journal of machine Learning research* 3 (1): 993-1022.

Boudin, Florian. 2016. "PKE: an open source python-based keyphrase extraction toolkit." *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, 69-73.

Boudin, Florian. 2018. “Unsupervised keyphrase extraction with multipartite graphs.” *arXiv preprint arXiv:1803.08721* Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies 2 (6): 667-672. 10.18653/v1/N18-2105.

Bougouin, Adrien, Florian Boudin, and Béatrice Daille. 2013. “Topicrank: Graph-based topic ranking for keyphrase extraction.” *ACL Anthology* Asian Federation of Natural Language Processing (10): 543-551. https://www.aclweb.org/anthology/I13-1062.

Brin, Sergey, and Lawrence Page. 1998. "The anatomy of a large-scale hypertextual web search engine."

Burns Kraft, Tristan F., Moul Dey, Randy B. Rogers, David M. Ribnicky, David M. Gipp, William T. Cefalu, Ilya Raskin, and Mary A. Lila. 2008. “Phytochemical composition and metabolic performance-enhancing activity of dietary berries traditionally used by native North Americans.” *Journal of agricultural and food chemistry* 56, no. 3 (1): 654-660. 10.1021/jf071999d.

Campos, Ricardo, Vítor Mangaravite, Arian Pasquali, Alípio M. Jorge, Célia Nunes, and Adam Jatowt. 2018. “YAKE! collection-independent automatic keyword extractor.” *European Conference on Information Retrieval*, (3), 806-810. https://doi.org/10.1007/978-3-319-76941-7_80.

Carter, Tina L., Kristin L. Morse, David W. Giraud, and Judy A. Driskell. 2008. “Few differences in diet and health behaviors and perceptions were observed in adult urban Native American Indians by tribal association, gender, and age grouping.” *Nutr Res.* 28, no. 12 (12): 834-841. 10.1016/j.nutres.2008.10.002.

Cho, Hae-Wol. 2019. “Topic Modeling.” *Osong public health and research perspectives* 10 (3): 115. https://doi.org/10.24171/j.phrp.2019.10.3.01.

Colby, Sarah E., Leander R. McDonald, and Greg Adkison. 2012. "Traditional Native American foods: stories from northern plains elders." *Journal of Ecological Anthropology* 15, no. 1: 65-73. 10.5038/2162-4593.15.1.5.

Corney, David P., Bernard F. Buxton, William B. Langdon, and David T. Jones. 2004. “BioRAT: extracting biological information from full-length papers.” *Bioinformatics* 20, no. 17 (7): 3206-3213.

Moerman. D.E., 2008. “Ethnobotany in Native North America.” *Encyclopaedia of the History of Science, Technology, and Medicine in Non-Western Cultures. Springer*. https://doi.org/10.1007/978-1-4020-4425-0.

Derczynski, Leon. . 2016. "Complementarity, F-score, and NLP Evaluation." In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 261-266

Duy Duc AnDD, Bui, Guilherme D. Fiol, and Siddhartha Jonnalagadda. 2016. “PDF text classification to leverage information extraction from publication reports.” *Journal of biomedical informatics* 61 (4): 141–148. 10.1016/j.jbi.2016.03.026.

El-Beltagy, Samhaa R., and Ahmed Rafea. 2010. “Kp-miner: Participation in semeval-2.” *Proceedings of the 5th international workshop on semantic evaluation*, (7), 190-193. https://www.aclweb.org/anthology/S10-1041.

Florescu, Corina, and Cornelia Caragea. 2017. “Positionrank: An unsupervised approach to keyphrase extraction from scholarly documents.” *In Proceedings of the 55th Annual*

Meeting of the Association for Computational Linguistics 1 (7): 1105-1115. 10.18653/v1/P17-1102.

Florescu, Corina, and Cornelia Caragea. 2017. "PositionRank: An Unsupervised Approach to Keyphrase Extraction from Scholarly Documents." *ACL Anthology* Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (7): 1105-1115. 10.18653/v1/P17-1102.

Goutte, Cyril, and Eric Gaussier. 2005. "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation." *European conference on information retrieval, Springer*, 345-359.

Ian, Witten H., Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig G. Nevill-Manning. 2005. "Kea: Practical automated keyphrase extraction." *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific*, 129-152.

Jonnalagadda, Siddhartha R., Pawan Goyal, and Mark D. Huffman. 2015. "Automating data extraction in systematic reviews: a systematic review." *Systematic reviews* 4 (1): 1-16. 10.1186/s13643-015-0066-7.

Kuhnlein, Harriet V., and Olivier Receveur. 1996. "Dietary change and traditional food systems of indigenous peoples." *Annual review of nutrition* 16 (1): 417-442.

Kwon, YI, E. Apostolidis, YC Kim, and K. Shetty. 2007. "Health benefits of traditional corn, beans, and pumpkin: in vitro studies for hyperglycemia and hypertension management." *J Med Food* 10, no. 2 (Jun): 266. 10.1089/jmf.2006.234. PMID: 17651062.

Ranilla, Lena Galvez, Apostolidis, Emmanouil, Genovese, Maria I, Lajolo, Franco M. and Shetty, Kalidas 2009. "Evaluation of indigenous grains from the Peruvian Andean region

for antidiabetes and antihypertension potential using in vitro methods." *Journal of medicinal food* 12, no. 4 (8): 704-713. 10.1089/jmf.2008.0122.

Li, Baoli, and Liping Han. 2013."Distance weighted cosine similarity measure for text classification." In *International Conference on Intelligent Data Engineering and Automated Learning*, pp. 611-618. Springer, Berlin, Heidelberg,.

Lin, Jimmy. 2009. "Is searching full text more effective than searching abstracts?." *BMC bioinformatics* 10, no. 1 (2): 46. 10.1186/147-2105-10-46.

Lipscomb, Carolyn E. 2000. "Medical subject headings (MeSH)." *Bulletin of the Medical Library Association* 88, no. 3 (7): 265.

Mahata, Debanjan, John Kuriakose, Rajiv R. Shah, and Roger Zimmermann. 2018. "Key2Vec: Automatic Ranked Keyphrase Extraction from Scientific Articles using Phrase Embeddings." *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 2 (6): 634-639. 10.18653/v1/N18-2100.

Marshall, Iain J., and Byron C. Wallace. 2019. "Toward systematic review automation: a practical guide to using machine learning tools in research synthesis." *Systematic reviews* 8, no. 1 (12): 163. https://doi.org/10.1186.s13643-019-1074-9.

Mihalcea, Rada, and Paul Tarau. 2004. "Textrank: Bringing order into text." *Proceedings of the 2004 conference on empirical methods in natural language processing*, 404-411.

Mishra, Lokesh K., Jacob Walker-Swaney, Dipayan Sarkar, and Kalidas Shetty. 2017. "Bioactive vegetables integrated into ethnic "Three Sisters Crops" garden targeting foods for type 2 diabetes-associated health disparities of American Indian communities." *Journal of Ethnic Foods* 4 (3): 163-171. https://doi.org/10.1016/j.jef.2017.08.007.

Nguyen, Thuy D., and Minh-Thang Luong. 2010. “WINGNUS: Keyphrase Extraction Utilizing Document Logical Structure.” *ACL Anthology* Proceedings of the 5th International Workshop on Semantic Evaluation (7): 166-169. https://www.aclweb.org/anthology/S10-1035.

Park, Sunmin, Nobuko Hongu, and James W. Daily III. 2016. “Native American foods: History, culture, and influence on modern diets.” *Journal of Ethnic Foods* 3, no. 3 (8): 171-177. https://doi.org/10.1016/j.jef.2016.08.001.

Patchell, Beverly, and Karethy Edwards. 2014. “The Role of Traditional Foods in Diabetes Prevention and Management among Native Americans.” *Current Nutrition Reports* 3, no. 4 (9): 340-344. https://doi.org/10.1007/s13668-014-0102-6.

Phillips, Katherine M., Pamela R. Pehrsson, Wanda W. Agnew, Angela J. Scheett, Jennifer R. Follett, Henry C. Lukaski, and Kristine Y. Patterson. 2014. “Nutrient composition of selected traditional United States Northern plains native American plant foods.” *Journal of Food Composition and Analysis* 34 (2): 136-152. http://dx.doi.org/10.1016/j.jfca.2014.02.010.

Pindus, Nancy, and Carol Hafford. 2019. “Food security and access to healthy foods in Indian country: Learning from the Food Distribution Program on Indian Reservations.” *Journal of Public Affairs* 19, no. 3 (2): e1876. https://doi.org/10.1002/pa.1876.

Rabby, Gollam, Saiful Azad, Mufti Mahmud, Kamal Z. Zamli, and Mohammed M. Rahman. 2018. “A Flexible Keyphrase Extraction Technique for Academic Literature.” *Procedia computer science* 135:553-563. 10.1016/j.procs.2018.08.208.

Ramos, Juan. 2003. “Using tf-idf to determine word relevance in document queries.” *Proceedings of the first instructional conference on machine learning* 242:133-142.

Ranilla, Lena G., Cinthya Huamán-Alvino, Orlando Flores-Báez, Edson M. Aquino-Méndez, Rosana Chirinos, David Campos, Ricardo Sevilla, et al. 2019. “Evaluation of phenolic antioxidant-linked in vitro bioactivity of Peruvian corn (Zea mays L.) diversity targeting for potential management of hyperglycemia and obesity.” *Journal of food science and technology* 56, no. 6 (6): 2909-2924. https://doi.org/10.1007/s13197-019-03748-z.

Saggion, Horacio, and Francesco Ronzano. 2017. “Scholarly data mining: making sense of scientific literature.” *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, (June), 1-2. 10.1109/JCDL.2017.7991622.

Sarkar, Dipayan, Jacob Walker-Swaney, and Kalidas Shetty. 2019. “Food diversity and indigenous food systems to combat diet-linked chronic diseases.” *Current Developments in Nutrition* 4, no. 1 (09): 3-11. https://doi.org/10.1093/cdn/nzz099.

Sterckx, Lucas, Thomas Demeester, Johannes Deleu, and Chris Develder. 2015. “"Topical word importance for fast keyphrase extraction.” *Proceedings of the 24th International Conference on World Wide Web*, (5), 121-122. https://doi.org/10.1145/2740908.2742730.

Thomas, James, Anna Noel-Storr, Iain Marshall, Byron Wallace, Steven McDonald, Chris Mavergames, Paul Glasziou, et al. 2017. “Living systematic reviews: 2. Combining human and machine effort.” *Journal of clinical epidemiology* 91 (9): 31-37. https://dx.doi.org/10.1016/j.jelinepi.2017.08.011.

Tokala Yaswanth Sri Sai, Santosh, Debarshi K. Sanyal, Plaban K. Bhowmick, and Partha P. Das. 2020. “DAKE: Document-Level Attention for Keyphrase Extraction.” *In European Conference on Information Retrieval* 12036 (4): 392-401. doi.org/10.1007/97https://doi.org/10.1007/978-3-030-45442-5_498-3-030-45442-5_49.

Tong, Zhou, and Haiyi Zhang. 2016. “A text mining research based on LDA topic modelling.” *International Conference on Computer Science, Engineering and Information Technology*, 201-210. 10.5121/csit.2016.60616.

Turney, Peter D. 2004. “Extraction of keyphrases from text: evaluation of four algorithms.” *arXiv preprint cs/0212014*, (12).

“United States Department of Agriculture.” n.d. Natural Resources Conservation Service. https://plants.sc.egov.usda.gov/java/.

Uysal, Alper Kursat, and Serkan Gunal. 2014. "The impact of preprocessing on text classification." *Information Processing & Management* 50, no. 1: 104-112.

Vermeulen, Andreas F. 2019. *Unsupervised Learning: Using Unlabeled Data*. Berkeley, CA: Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-5316-8_6.

Wan, Xiaojun, and Jianguo Xiao. 2008. “CollabRank: towards a collaborative approach to single-document keyphrase extraction.” *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, (8), 969-976. https://www.aclweb.org/anthology/C08-1122.

Warne, Donald, and Siobhan Wescott. 2019. “Social determinants of American Indian nutritional health.” *Current Developments in Nutrition* 3, no. 2 (8): 12-18.

Westergaard, David, Hans-Henrik Stærfeldt, Christian Tønsberg, Lars J. Jensen, and Søren Brunak. 2018. “A comprehensive and quantitative comparison of text-mining in 15 million full-text articles versus their corresponding abstracts.” *PLoS computational biology* 14, no. 2 (2): 1-16. https://doi.org/10.1371/journal.pcbi.1005962.

Willard, Dan E. 1984. “New trie data structures which support very fast search operations.” *Journal of Computer and System Sciences* 28 (3): 379-394.

Zamora-Kapoor, A., K. Sinclair, L. Nelson, H. Lee, and D. Buchwald. 2019."Obesity risk factors in American Indians and Alaska Natives: a systematic review." *Public health* 174: 85-96