2-2019

# Automatic Extraction of Useful Information from Food -Health Articles related to Diabetes, Cardiovascular Disease and Cancer

Ken Sunong
*Sheridan college*

# AUTOMATIC EXTRACTION OF USEFUL INFORMATION FROM FOOD-HEALTH ARTICLES RELATED TO DIABETES, CARDIOVASCULAR DISEASE AND CANCER

A Thesis

Presented to

The Faculty of Applied Science and Technology, School of Applied Computing

of

Sheridan College, Institute of Technology and Advanced Learning

by

SUONG, KEN

In partial fulfillment of requirements

for the degree of

Bachelor of Applied Computer Science – Mobile Computing

February 2019

ABSTRACT

# AUTOMATIC EXTRACTION OF USEFUL INFORMATION FROM FOOD-HEALTH ARTICLES RELATED TO DIABETES, CARDIOVASCULAR DISEASE AND CANCER

**Ken Suong**
**Sheridan College, 2019**

**Supervisor:**
**Dr. El Sayed Mahmoud**

Food-health articles (FHA) contain invaluable information for health promotion. However, extracting this information manually is a challenging process due to the length and number of articles published yearly. Automatic text summarization efficiently identifies useful information across large bodies of text which in turn speeds up the delivery of useful information from FHA. This research work aims to investigate the performance of statistical based summarization and graphical based unsupervised learning summarization in extracting useful information from FHA related to diabetes, cardiovascular disease and cancer. Various combinations of introduction, result and conclusion sections of three hundred articles were collected, preprocessed and used for evaluating the performance of the two summarization technique types. Generated summaries are compared to the original abstracts using two measures. The first quantifies the similarity of the generated summary to the abstract. The second measure gauges the coverage of the generated summary and the article abstract to the article sections. Overall, this experiment showed the automatically generated summaries are not comparable to the human-made abstracts found in FHA and there is room for improvement since the highest similarity of the generated to the written abstract was 52-57% and the sentence scoring of summarization could be optimized for various domains.

## *ACKNOWLEDGEMENTS*

I'd like to acknowledge Dr. El Sayed Mahmoud for his efforts in guiding the research process and help in putting together this proposal; Dr. El Sayed Abdelaal, research scientist, for supplying the training data for the thesis; Dr. Edward Sykes, Dr. Aeiman Gadafi, and Dr. Yuchong Rachel Jiang for their review and feedback of the thesis. I'd like to extend my acknowledgement to my parents and to anyone who helped me to get where I am today.

***TABLE OF CONTENTS***

## *LIST OF TABLES*

## *LIST OF FIGURES*

# *CHAPTER ONE*

# *INTRODUCTION*

## *1.1 The Problem Context*

Food-health articles (FHA) are important for the general public and for stakeholders involved in healthcare because they contain invaluable information for health promotion. Reading FHA would help people become more knowledgeable about the impact that food has on their body. This will help people make healthier food choices for themselves which in turn promote public health. However, reading a single FHA consumes a significant amount of time because of the large number of FHA pages with multiple sections of text per page. Additionally, choosing the most relevant FHA is a time-consuming process because many FHA are published daily and to go through all of those articles is a difficult task for anyone and getting the necessary information that they need is even more difficult (Ross et al, 2018).

Text summarization can play a significant role in speeding up the delivery of food health knowledge to the public by generating a short summary for the FHA without ignoring important pieces of information. The growing trend of publishing FHA on the internet increases the value and the need for automatic summarization. Abstracts from FHA summarize published articles and are written by the authors. Writing an abstract of a paper requires familiarity with the paper and the subject matter of the paper as well (Lloret et al, 2013). The ability to bring out the necessary content from an article and condense it with limited words requires skills (Luhn, 1958). Abstracts generated from authors can also be influenced by a writer's attitude and their interpretation of the article can be biased and can give an inaccurate retelling of the article (Luhn, 1958). This work

investigated two summarization approaches for generating a short summary for the FHA(s) related to diabetes, cardiovascular disease (CVD), and cancer.

### *1.2 Terms and Definitions*

| Term | Definition |
|---|---|
| Natural Language Processing (NLP) | Area of research and application that explores how computers can be used to analyze, understand, and manipulate natural text or speech for useful applications. |
| Sentiment Analysis | Used to identify the feeling, opinion, or belief of a statement. |
| Summarizer | Summarize a block of text, exacting topic sentences, and ignoring the rest |
| Natural Language Toolkit (NLTK) | Python library providing modules for processing text, classifying, tokenizing, stemming, tagging, and parsing text |
| N-grams | A continuous sequence of n items from a given sample of text and speech. N-grams are collected from the text in scientific articles |

### *1.3 The Problem Statement*

There are over 150,000 papers every year for the past 18 years that relate nutrition to the diseases: diabetes, CVD and cancer (Figure 1). This makes it difficult for both health professionals and patients to extract useful information from the papers related to their interest. This work investigated two different text summarization approaches for generating a short, clear and complete summary of FHA.

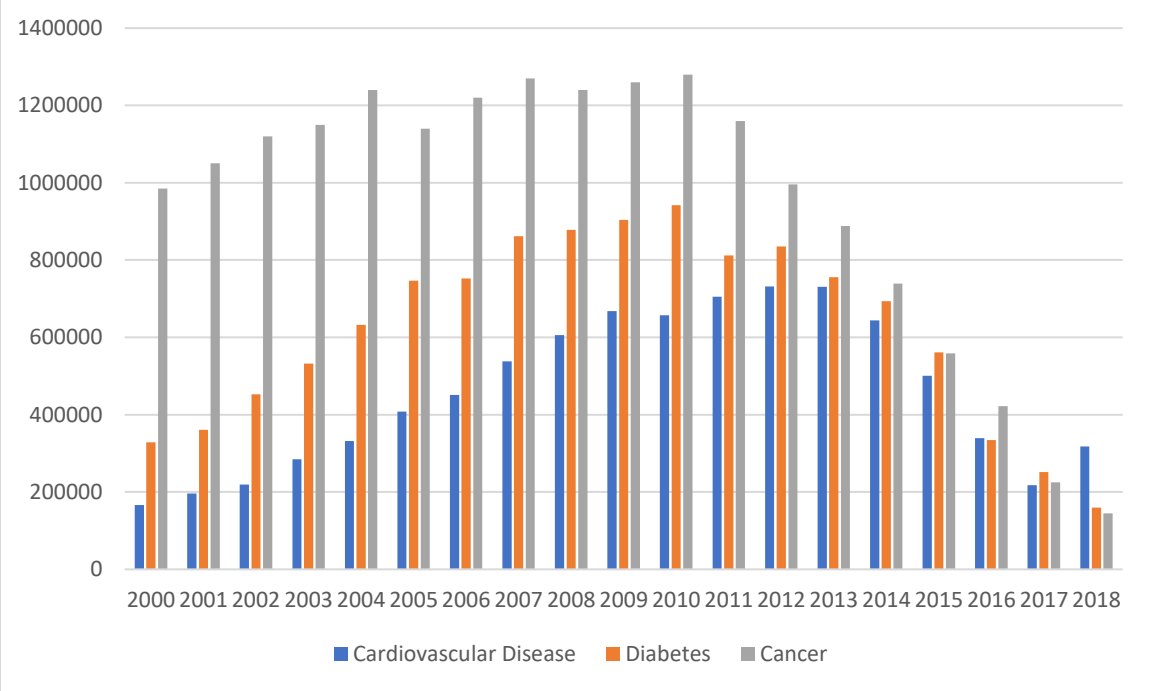



**Figure 1. The number of papers at different years using cardiovascular disease, diabetes, and cancer as keywords in Google Scholar.**

### *1.4 The Purpose*

The purpose of this work is to examine the potential for two different types of summarization approaches to generate an effective and short summary for the FHA(s) related to the diseases: diabetes, CVD and cancer. The first summarization approach is

statistics-based and the second is machine learning-based. The two approaches were applied to various combinations of the FHA sections such as introduction combined with result or introduction combined with methodology and results. The thesis developed a similarity metric to evaluate the similarity of the resulting summary relative to the paper abstract written by the paper author. The ultimate goal of this research is to provide guidelines and tools that improves the efficiency of automatic information extraction from FHA related to the diseases: diabetes, CVD and cancer.

The performance of the selected summarization techniques was quantified using a similarity metric that measures the similarity of the resulting summary to the abstract written by the FHA author and was applied to various combinations of the FHA sections such as introduction combined with result or introduction combined with results and conclusion. This metric measures similarity based on n-grams. The coverage of the automatically generated summary to the different sections of the article is measured using the variance of the number of sentences in the summary belong to each section.

### *1.5 Motivation*

Text summarization is important because large bodies of text need to be summarized to something that can be easily consumed by the reader. This thesis aims to evaluate different summarization methods and see how well they perform and to tell the user which summarization method work better for summarizing FHA. The summarization methods are statistical based summarization and graphical based unsupervised learning algorithm summarization.

Beyond technology and NLP, this thesis also aims to improve the way people interpret food health articles. FHA articles are available to researchers in the scientific community but the sheer number of articles that they have to read is immense. Having a good summarizer can also accurately summarize FHA articles which do not have abstracts and can help them save time and help them focus on their research more.

This research aligns with the Government of Canada's vision of the agriculture-food sector of Canada to promote safety, sustainability and high quality of food products (Report of Canada's Economic Strategy Tables: Agri-Food, 2018).

### *1.6 The Proposed Work*

The information provided in the science articles is summarized using a statistical text summarization approach and a machine learning based approach. The summaries were generated based on various combinations of the FHA sections such as introduction combined with results or introduction combined with the methodology. The summaries are then compared to the abstracts to evaluate the performance of the summarization approaches based on the similarity of their summaries to the abstracts written by the authors and the coverage of the summaries and the abstracts to the article sections. The similarity between a generated summary and the corresponding abstract was measured by comparing the n-grams terms for the summary to the corresponding abstract. An automatic comparison was performed between the summaries and the corresponding abstracts to evaluate the coverage of the generated summary compared to the coverage of the corresponding abstracts to the different sections of the article. This experiment used 300 papers from cancers, diabetes, and cardiovascular diseases. The generated summaries

and the abstracts of the 300 FHA articles have been analyzed to evaluate the ability of the selected summarization algorithms to extract useful information from the FHA(s) related to the three diseases.

Two approaches to automatic text summarization were used: statistical based summarization tools and graphical based unsupervised learning algorithm summarization tools using a tool called LexRank (Liang et al, 2012). These two were chosen to compare their effectiveness when generating summaries for scientific articles based on their success in text summarization literature as well as their popularity and their ease of use. LexRank uses unsupervised learning for text summarization using graph-based centrality to score sentences (Liang et al, 2012). The graph maps all the sentences from a body of text and will recommend sentences to be used in the summary based on similarity to other sentences (Liang et al, 2012). Similar sentences are seen as important and sentences that are recommended will also be seen as important which will get the sentence ranked more highly which will have a greater chance of being placed in the summary (Liang et al, 2012).

With advances of text summarization techniques and the application of extraction-based summarization, this research hopes to summarize scientific articles accurately and efficiently.

### *1.7 Thesis statement*

A text summarization system can be developed that is able to automatically generate a summary for food-health articles relevant to the three proposed diseases such that the generated summary contains information comparable to the article abstract

written by the author. The proposed summary approaches evaluate the significance of each sentence in the articles and use the most significant sentences to generate the summary. This research aims to determine the summarization approach including relevant settings that extracts comparable useful information to the information presented by the abstract written by the author.

### *1.8 Contributions*

This work showed how to use summarization approaches to generate a summary for food-health articles related to the diseases: CVD, diabetes and cancer. The contributions of this work include:

- Identified the best sections in the FHA to be used as a source for the summary by showing that including the introduction, results, and conclusion would generate better summaries than any combinations
- Developed a measure that quantifies the similarity between the generated summary and the abstract written by the author.
- Developed a measure that quantifies the coverage of the generated summary to the article sections.

### *1.9 Organization of Thesis*

The remainder of this thesis consists of a literature review, methodology and results. The literature review focuses on how NLP currently summarizes text, why tokenization is important for NLP and text summarization, and the different approaches in text summarization. The methodology section describes the details of the methodologies

involved in the work. This includes the pipeline for producing the summaries, how Natural Language Tool Kit (NLTK) will be used to extract words from text, how tokens will be used for the summarization process. NLTK is a Python program that has tools to work with human language and can be found in https://www.nltk.org/. The results section highlights the experimental findings including the analysis of these findings. The conclusion summarizes the experimental findings, explains the impact of automatic summarizers with respect to FHA, and potential future research.

# *CHAPTER TWO*

# *CONTEXTUAL BACKGROUND AND LITERATURE REVIEW*

Text summarization is a significant process that can accelerate the knowledge delivery to the public when the summary contains the useful information in the source text. This work focuses on extracting useful information from FHA(s) related to diabetes, CVD and cancer. This chapter reviews relevant research to text summarization including steps of text summarization approaches, types of approaches (i.e. statistical and machine-learning based) and how these different approaches generate the summary.

## *2.1. NLP for text summarization*

Bui et al, 2016, have developed a text summarization system was created to gather data from full text in systematic review development (Bui et al., 2016). They extracted data from publication reports in a standard process in systematic review development and developed a text summarization system aimed at enhancing productivity and reducing errors in the traditional data extraction process. They used machine learning and NLP to generate summaries of full-text scientific publications and attempted to summarize clinical data elements like sample size, group size, and PICO values (Bui et al., 2016). Computer-generated summaries compared with human-written summaries (title and abstract) and looked for the presence of necessary information for the data extraction and were able to produce summaries that covered more information than the summaries created by humans (Figure 2).

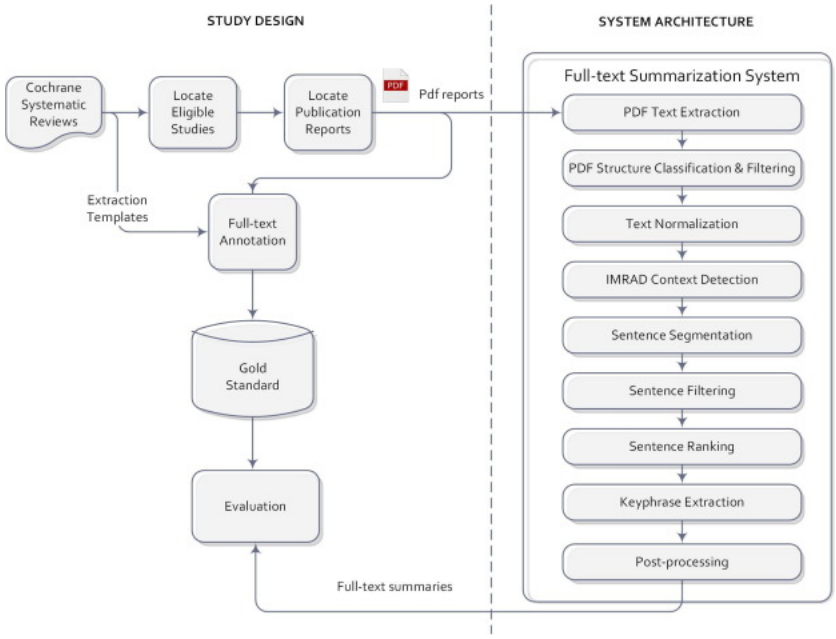



**Figure 2: Overall architecture of text summarization system for articles (Bui et al., 2016).**

Another experiment used NLP for spoken diet records in order to focus on dietary assessment (Lacson et al., 2006). Previous methods of dietary assessment include written records, 24-hour recalls, and food frequency questionnaires and attempted to use mobile phones provide real-time dietary records instead of written records (Bui et al., 2016). Understanding a perfect transcript of spoken dietary records is challenging and the approach takes the identification of food items, identification of food quantifiers, classification of food quantifiers and temporal annotation. They proposed a method for automatically processing transcribed SDRs and used natural language to evaluate what they ate and the density of relevant sentences (Bui et al., 2016).

### *2.1.1. The Use of Tokenization in NLP*

The tokenization process involves breaking up a body of text into basic units called tokens. This process is tokenization omits characters like punctuation and tokenization is commonly used as the first step in NLP and in automatic summarization (Hassler and Fliedl, 2006). Tokenization is an important step for text summarization because it allows text mining of large text, the ability to assess each word or group of words individually and determine specific patterns based on classification of the words (Hassler and Fliedl, 2006). The additional feature of using tokens instead is that tokens can be language independent and can be used by NLP algorithms for performing pattern matching and categorization (Hassler and Fliedl, 2006).

## *2.2. Extractive and Abstractive Summarization*

Extractive and abstractive summarization are two common methods for text summarization. Extractive summarization involves taking important sentences, words, and paragraphs from a document and transforming them into a shorter form. Terms that are deemed important are decided based on statistical and linguistic features (Gupta, 2010). Extractive text summarization has been successful on multi-document datasets (Varalakshmi and Kallimani, 2018).

Abstractive summarization involves understanding the main concepts from a document and then making those concepts in natural language. Linguistic methods are used to interpret the document and generate expressions that would best describe the interpretation in the form of shorter text and conveys the most important information. (Gupta, 2010). Abstractive text summarization techniques have also been successful on multi-document datasets as well (Raphal, et al, 2018).

### *2.3 Statistical Based Approaches to Text Summarization*

Statistical Based Approach involves the extraction of keywords from a document. The keywords extracted go through statistical features to determine the characteristic of the document such as word frequency, term frequency-inverse document frequency, and position of the keyword (Webster and Kit, 1992). Chandra et al, 2011 used a summarization approach by extracting the most essential concepts with text mining techniques. The research developed a statistical automatic text summarization approach using a probabilistic model in order to improve the performance of the summaries. The term weights are determined using a probabilistic model and then identifies the relationships to determine the semantic relationship significance of nouns. The better the semantic relationship significance value is, the better the rank score for the sentence.

To determine the significance of the sentences in an article, the words of these sentence are analyzed. The frequency of a word occurrences in an article would indicate that this word is a significant word. The relative position a word within a sentence can also be used as a useful measurement for determining the significance of the sentence. Significance is based on those two measurements (Chandra et al, 2011).

A writer normally repeats certain words when elaborating on a certain subject and it indicates more emphasis and thus the word is more significant (Chandra et al, 2011).This scoring does not differentiate between word forms. Thus, words with different tenses are considered identical and are considered the same word. Inventory of the words is taken to generate a word list and frequency of those words is taken. The procedure for this is simple and is not computationally complex. Authors use different words to describe the same thing is unlikely and, in the event, that authors use synonyms for stylistic reasons,

the authors will likely run out of alternatives (Chandra et al, 2011). Automatically generated abstracts have high-degree of reliability, consistency, and stability because they do not have the variations and orientation of human capabilities and are generated using statistical analysis using the authors own words (Chandra et al, 2011).

### *2.4. Machine Learning Based Approaches for Text Summarization*

Machine learning based approach requires features and an annotated dataset to train the models. Most popular machine learning techniques include Naïve Bayes, decision trees, Hidden Markov Model, Neural Network and Support Vector Machines. (Webster and Kit, 1992).

Machine learning has also had an impact on text summarization. It closely resembles classification problems where the training models are the "summary sentence" when they belong to the reference summary or "non-summary sentence" if they are not. Naïve Bayes and Neural Networks are machine learning methods used to generate summaries (Kumar, 2016). Machine learning based approaches use unsupervised and supervised learning methods to perform text summarization.

Unsupervised learning methods do not need to learn from premade human summaries and will attempt to decide the most important features from a document. Approaches that used unsupervised methods include graph-based, concept-based, fuzzy logic-based, and latent semantic analysis (Moratanch & Chitrakala, 2017).

The graph-based approach uses graphs to represent a document (Yang, 2018). The nodes in the graph comprise of different features found in the document and have iterative ranking for each node helps in determining important sentences and building

coherent final summaries (Kaynar et al, 2017). LexRank uses a graph-based approach to determine the salience of a sentence using Eigenvector centrality. LexRank breaks down the document into graph nodes that contain sentences and the edges between each sentence is the weighted cosine similarity values. Sentences of similar weight are clustered together into groups and those sentences are ranked using a LexRank scoring algorithm (Moratanch & Chitrakala, 2017).

Fuzzy logic-based approach uses a defuzzifier, fuzzifier, fuzzy knowledge base and inference engine to determine if sentences in a document are significant. The fuzzy system will take a document to extract features from it. The order in which the sentences occur in the original document and the ranking of the sentences based on fuzzy logic will generate the summary (Moratanch & Chitrakala, 2017).

The concept-based approaches use an external knowledge source like Wikipedia to extract concepts. Sentences are extracted from a document and are ranked based on importance. The rank is calculated using a conceptual vector or graph model to compare the concepts from the external knowledge source to the document sentences and similar sentences are eliminated to reduce redundancy in the final summary (Moratanch & Chitrakala, 2017).

Supervised learning methods use pre-made human summaries to learn and classify summary and non-summary sentences. A human is needed to label what sentences are summary and non-summary sentences (Moratanch & Chitrakala, 2017).

The Naïve Bayes classifier is fed data from a document for learning and makes features independent from each other. The probability of being included in the summary is determined by the number of features in the sentence. The probability will be used to

score the sentence and the highest scoring sentences will be used in the summary. Naïve Bayes rule has a training stage that takes in training documents and extractive summaries and sentences are then classified as either non-summary or summary based on features in the sentence. The classification is learned from the training data based on Bayes rule which uses the set of sentences and the features used the classification stage. Based on those features, Bayes rule will give a probability to how likely the sentence will be included in the summary (Moratanch & Chitrakala, 2017).

The Neural Network approach involves using neural nets to determine what sentences are important in a document (Zhong et al, 2015). RankNet is an algorithm developed by Burges et al. that is used in conjunction with a two-layer neural network and backpropagation. Training data is labeled and then features are extracted from the sentences in both test and training sets. The neural net takes in the sentences for ranking. Another proposed approach involves a three-layered feed-forward neural net and learns the characteristics of what summary and non-summary sentences are. Infrequent features are eliminated and frequent features are brought together to rank sentences and determine which sentences are important (Moratanch & Chitrakala, 2017).

This work uses statistical-based approaches and graph-based approaches for generating text summaries and both are used because of their simplicity, easy implementation, and usefulness.

## *CHAPTER THREE*

## *METHODOLOGY*

This chapter discusses the methodology used in this research work. This includes the steps for building the article summarization system, measures, testing and evaluation strategies.

### *3.1 The proposed summarization system*

This work aims to identify the components of a text summarization system for FHA(s) related to diabetes, CVD and Cancer. This includes answering two questions. The first is what are the parts of the article that provide more information for generating a summary? The second question is which type of summarization techniques is appropriate for summarizing FHA(s) of these three diseases. The proposed methods were designed to answer these two questions and the methods are shown in Figure 3. The figure shows the data fed to the summarization approach (summarizer) including the testing strategy for the summarizer's output.

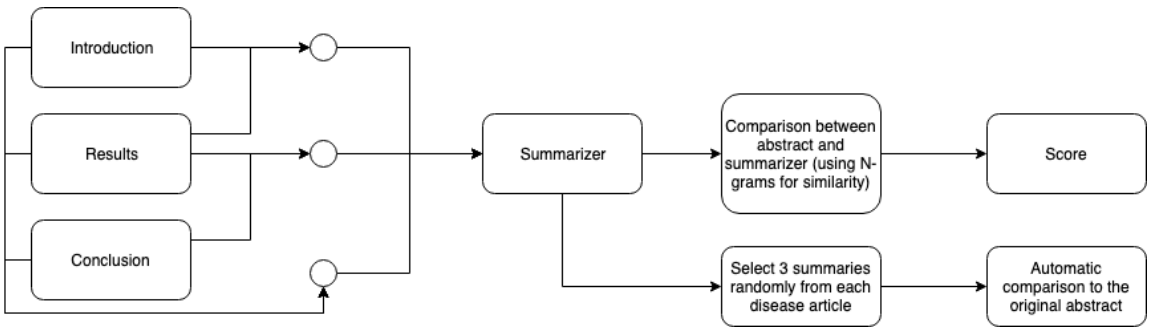



**Figure 3: The overall method of producing and evaluating summaries from papers.**

### *3.1.1 Data*

One hundred articles per each disease have been used to develop and test the proposed summarization system. Each article consists of different sections such as introduction and conclusion. This work focuses on using various combinations of introduction, results, and conclusions because this work aims to identify the significance of the different section combinations to the performance of the generated summary. Three combinations were fed to the summarizer (text summarization approach). These combinations are introduction-results; results-conclusions; and introduction-results-conclusion. Each combination contains bodies of text which went through a preprocessing phase where extra whitespace and newline characters are removed. Other items that are irrelevant such as citation referencing using numbers are also removed. Each category had one hundred articles and the text from each article was extracted by copying the text and placing the text in a text file.

Data preprocessing was performed using Python. The first step in preprocessing was to removing large white spaces found between paragraphs and newline characters by converting them into regular spaces or None if found at end of the text. The code snippet below finds the Unicode for form feed page breaks (\f), a horizontal tab that makes indents for the beginning of each paragraph (\t), line feed that makes paragraphs go to next line(\n), and carriage return that may be found at the end of the text (\r). A regex expression was also used to remove contents with square brackets since those contents are typically citations made by the author and would not be helpful for summarizing the text.

```
    replace = {
    ord('\f') : ' ',
    ord('\t') : ' ',
    ord('\n') : ' ',
    ord('\r') : None
    }
```

### *3.1.2 Tokenization and Frequency Count Processes*

After preprocessing the data, the bodies of text were tokenized. Tokenization process breaks the received text into sentences and words. The Natural Language Tool Kit (NLTK) provides several useful tools for the tokenization process. Stop words need to be removed in order to not have an impact on the scoring of the sentences on the final summary. NLTK has a list of stop words (stopwords from nltk.corpus) and removed those words from the text as well as punctuation. NLTK was also used to turn words into lower case and to return unique words from the input (word_tokenize from nltk.tokenize). The tokenized content was stored into individual sentences (sent_tokenize from nltk.tokenize) while words that are not in stop-word list or punctuation were returned. The frequency of the words was stored to be used later by the summarizer to identify important words and sentences.

```
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from string import punctuation
stop_words = set(stopwords.words('english') + list(punctuation))
words = word_tokenize(content.lower())

return [
    sent_tokenize(content),
    [word for word in words if word not in stop_words]
]
```

### *3.1.3 Summarizer*

In order to generate a summary of an article, a scoring system was set up. Having a list of unique sentences and unique words generated in the previous step (tokenization), a score was generated to determine the frequency of each word occurring in the text and use that to assign a score for each sentence using FreqDist function from nltk.probability.

```
word_freq = FreqDist(word_tokens)

# initialize a defaultdict for making a frequency map
ranking = defaultdict(int)

# iterate over the sentences and increase score based on frequency
for i, sentence in enumerate(sentence_tokens):
    for word in word_tokenize(sentence.lower()):
        if word in word_freq:
            ranking[i] += word_freq[word]
```

The frequency of each of the filtered words from the list of tokens was determined and then sentences were iterated over and the rank of each sentence ranking went up based on the frequency of the words of this sentence found from the list of tokens. An example of the summarization process can be found in Figure 4a, 4b, 4c, and 4d.

```
0: It is well accepted that obesity increases the risk of type 2 diabetes.
1: Risk of diabetes may be influenced by the severity of obesity, duration of obesity, and magnitude and rate of weig
ht gain.
2: It is unclear what level of detail is required to best capture the health risks associated with obesity in a clini
cally practical and meaningful way.
3: It has been argued that examination of duration of obesity is important as, unlike measurements of body mass index
(BMI) at one point in time, it takes into account any cumulative effect of obesity.
4: Being obese for longer is associated with an increased risk of type 2 diabetes.
5: More recently, a construct known as obese-years, which takes into account both how long a person has been obese an
d their magnitude of obesity, was shown to be a better predictor of diabetes risk than duration of obesity or level o
f BMI alone.
6: However, the computation of obese-years requires knowledge of BMI regularly throughout life, and is therefore less
straightforward than measures such as BMI at a single point in time or duration of obesity alone.
7: This potentially limits the use of the obese-years metric in practical situations.
8: There is limited data on the relationship between age of onset of obesity and risk of diabetes.
9: There is evidence that men who were overweight in early adulthood were more likely to develop type 2 diabetes in m
iddle-to-older age.
10: Age of onset may represent a crude marker for duration of obesity, particularly in settings where people rarely r
evert from an obese state to a non-obese state.
11: Age of onset of obesity, alone or combined with current BMI, may provide sufficient information on health risks a
ssociated with obesity and represent a simpler alternative to the obese-years metric as a predictor of type 2 diabete
s.
12: The aim of this study was to compare age of onset of obesity with the obese-years construct as predictors of risk
of type 2 diabetes.
13: In this large cohort, the obese-years construct performed better than BMI alone and age of onset of obesity plus
BMI in explaining the observed information on risk of incident type 2 diabetes.
14: In contrast to our expectations a model with current BMI and age of onset of obesity did not appear to improve pr
ediction of type 2 diabetes compared to a model with BMI alone, except in the elderly.
15: This study is novel in that no previous study has examined age of onset of obesity in relation to risk of type 2
diabetes.
16: Our findings indicate that while obese-years was the optimal construct to predict risk of type 2 diabetes, age of
onset of obesity in combination with current BMI might be better than BMI alone in the elderly.
17: As the calculation of obese-years is less straight forward than age of onset of obesity, the latter may be a usef
ul practical alternative although at the cost of reduced ability to explain diabetes incidence.
18: The strengths of this study include a long period of follow-up with biennial examinations and measured height and
weight.
19: The study was limited by the modest number of participants who developed obesity over the follow-up period (n=590
(14%)).
20: Additionally, as it was not possible to include any obesity duration prior to the commencement of the study, tota
l obesity duration may be underestimated.
21: In conclusion, this study demonstrated that the obese-years construct was the optimal construct to predict risk o
f type 2 diabetes, indicating the importance of both duration of obesity and the level of obesity in the risk of deve
loping diabetes.
```

**Figure 4a) A document with the introduction and conclusion containing 21 sentences.**

```
['well', 'accepted', 'obesity', 'increases', 'risk', 'type', '2', 'diabetes', 'risk', 'diabetes', 'may', 'influence
d', 'severity', 'obesity', 'duration', 'obesity', 'magnitude', 'rate', 'weight', 'gain', 'unclear', 'level', 'detai
l', 'required', 'best', 'capture', 'health', 'risks', 'associated', 'obesity', 'clinically', 'practical', 'meaningfu
l', 'way', 'argued', 'examination', 'duration', 'obesity', 'important', 'unlike', 'measurements', 'body', 'mass', 'in
dex', 'bmi', 'one', 'point', 'time', 'takes', 'account', 'cumulative', 'effect', 'obesity', 'obese', 'longer', 'assoc
iated', 'increased', 'risk', 'type', '2', 'diabetes', 'recently', 'construct', 'known', 'obese-years', 'takes', 'acco
unt', 'long', 'person', 'obese', 'magnitude', 'obesity', 'shown', 'better', 'predictor', 'diabetes', 'risk', 'duratio
n', 'obesity', 'level', 'bmi', 'alone', 'however', 'computation', 'obese-years', 'requires', 'knowledge', 'bmi', 'reg
ularly', 'throughout', 'life', 'therefore', 'less', 'straightforward', 'measures', 'bmi', 'single', 'point', 'time',
'duration', 'obesity', 'alone', 'potentially', 'limits', 'use', 'obese-years', 'metric', 'practical', 'situations',
'limited', 'data', 'relationship', 'age', 'onset', 'obesity', 'risk', 'diabetes', 'evidence', 'men', 'overweight', 'e
arly', 'adulthood', 'likely', 'develop', 'type', '2', 'diabetes', 'middle-to-older', 'age', 'age', 'onset', 'may', 'r
epresent', 'crude', 'marker', 'duration', 'obesity', 'particularly', 'settings', 'people', 'rarely', 'revert', 'obes
e', 'state', 'non-obese', 'state', 'age', 'onset', 'obesity', 'alone', 'combined', 'current', 'bmi', 'may', 'provid
e', 'sufficient', 'information', 'health', 'risks', 'associated', 'obesity', 'represent', 'simpler', 'alternative',
'obese-years', 'metric', 'predictor', 'type', '2', 'diabetes', 'aim', 'study', 'compare', 'age', 'onset', 'obesity',
'obese-years', 'construct', 'predictors', 'risk', 'type', '2', 'diabetes', 'large', 'cohort', 'obese-years', 'constru
ct', 'performed', 'better', 'bmi', 'alone', 'age', 'onset', 'obesity', 'plus', 'bmi', 'explaining', 'observed', 'info
rmation', 'risk', 'incident', 'type', '2', 'diabetes', 'contrast', 'expectations', 'model', 'current', 'bmi', 'age',
'onset', 'obesity', 'appear', 'improve', 'prediction', 'type', '2', 'diabetes', 'compared', 'model', 'bmi', 'alone',
'except', 'elderly', 'study', 'novel', 'previous', 'study', 'examined', 'age', 'onset', 'obesity', 'relation', 'ris
k', 'type', '2', 'diabetes', 'findings', 'indicate', 'obese-years', 'optimal', 'construct', 'predict', 'risk', 'typ
e', '2', 'diabetes', 'age', 'onset', 'obesity', 'combination', 'current', 'bmi', 'might', 'better', 'bmi', 'alone',
'elderly', 'calculation', 'obese-years', 'less', 'straight', 'forward', 'age', 'onset', 'obesity', 'latter', 'may',
'useful', 'practical', 'alternative', 'although', 'cost', 'reduced', 'ability', 'explain', 'diabetes', 'incidence',
'strengths', 'study', 'include', 'long', 'period', 'follow-up', 'biennial', 'examinations', 'measured', 'height', 'we
ight', 'study', 'limited', 'modest', 'number', 'participants', 'developed', 'obesity', 'follow-up', 'period', 'n=59
0', '14', 'additionally', 'possible', 'include', 'obesity', 'duration', 'prior', 'commencement', 'study', 'total', 'o
besity', 'duration', 'may', 'underestimated', 'conclusion', 'study', 'demonstrated', 'obese-years', 'construct', 'opt
imal', 'construct', 'predict', 'risk', 'type', '2', 'diabetes', 'indicating', 'importance', 'duration', 'obesity', 'l
evel', 'obesity', 'risk', 'developing', 'diabetes']
```

**Figure 4b) The words in the sentences were tokenized and turned into lower case. NLTK has a list of stop words that are removed to reduce the impact on the scoring of the sentences on the final summary.**

```
defaultdict(<class 'int'>, {0: 73, 1: 95, 2: 45, 3: 86, 4: 54, 5: 137, 6: 86, 7: 18, 8: 73, 9: 53, 10: 73, 11: 157, 1
2: 114, 13: 144, 14: 122, 15: 107, 16: 148, 17: 90, 18: 22, 19: 43, 20: 84, 21: 168})
```

**Figure 4c) A frequency map based on the filtered list of words and was used to produce a map of each sentence its total score. The frequency of each word that occurred in the text is used to grade the sentences.**

```
'Age of onset of obesity, alone or combined with current BMI, may provide sufficient information on health risks asso
ciated with obesity and represent a simpler alternative to the obese-years metric as a predictor of type 2 diabetes.
In this large cohort, the obese-years construct performed better than BMI alone and age of onset of obesity plus BMI
in explaining the observed information on risk of incident type 2 diabetes. Our findings indicate that while obese-ye
ars was the optimal construct to predict risk of type 2 diabetes, age of onset of obesity in combination with current
BMI might be better than BMI alone in the elderly. In conclusion, this study demonstrated that the obese-years constr
uct was the optimal construct to predict risk of type 2 diabetes, indicating the importance of both duration of obesi
ty and the level of obesity in the risk of developing diabetes.'
```

**Figure 4d) The final summary. Sentences 11, 13, 16, and 21 were used because they had the highest scores.**

The summary generated was configured to contain only four sentences in order to generate summaries that are concise and relevant. The four sentences are selected based on the sentence score calculated and stored in the tokenization process. The sentences have been sorted in descending order. The sorted list of the sentences has been transformed into a list of numeric positions. Each sentence gathered from the tokenized list is placed into the final summary and made sure that they appear in a logical order (introduction comes first, results in the middle, the conclusion comes last). This created the summary which is the product of the summarizer. Two types of summarizer were investigated in this study. The first is a statistically based summarizer and the second is machine-learning-based and called LexRank.

LexRank generates a graph that contains all the sentences in a document. Each sentence is a node in the graph, the edges are similarity relationship between sentences. LexRank uses a bag-of-words model to measure the similarity between sentences and similarity between sentences is determined by the frequency of word occurrence in a sentence. It uses the TF-IDF formulation where TF is term frequency and IDF is Inverse Document Frequency. It calculates the TF results in similarity strength when there are more word occurrences. IDF takes low occurrence words and how they inversely contribute to a higher value to the measurement. The magnitude of similarity between sentences is calculated using a combination of TF-IDF and cosine similarity in the IDF-modified-cosine formula (Figure 5).

$$\text{idf-modified-cosine}(x, y) = \frac{\sum_{w \in x,y} \text{tf}_{w,x}\text{tf}_{w,y}(\text{idf}_w)^2}{\sqrt{\sum_{x_i \in x}(\text{tf}_{x_i,x}\text{idf}_{x_i})^2} \times \sqrt{\sum_{y_i \in y}(\text{tf}_{y_i,y}\text{idf}_{y_i})^2}}$$

**Figure 5. The IDF-modified-cosine formula of LexRank (Erkan and Radev 2004).**

The formula measures the magnitude between sentences. Two sentences are similar if they are closer to each other which is determined when the cosine angle between sentences is smaller (Figure 6).

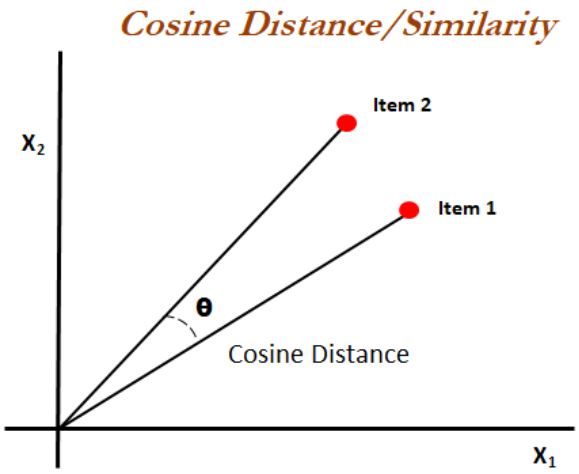



**Figure 6. Cosine distance/similarity (Dangeti, 2017)**

The calculated similarity is used to build a similarity matrix which can be used in a similarity graph. The LexRank algorithm analyzes the graph and the sentences that make up the nodes in the graph and the importance those sentences have to the neighbouring sentences.

Important sentences are filtered out of the similarity matrix using a thresholding mechanism. A subset of the similarity graph is generated and nodes that have the highest

degree of similarity are chosen as the sentence that represents the summary of the sentence.

### *3.2 Measures*

Two measures were developed to quantify the similarity and the coverage performance of the generated summary compared to the abstract written by the FHA author. The first measure is called *Similar N-grams* and the second called *Summary* coverage. The two measures are described in the following two sections.

#### *3.2.1 Similar N-grams measure*

In this research, a new similarity measure called Similar N-grams was developed to quantify the similarity between the generated summary and the article abstract based on the number of similar n-gram terms in both of the generated summary and the abstract written by the FHA author. This measure tokenizes the summary and the abstract using the n-gram into two separate lists of n-gram terms. The first contains the n-grams of the summary and the second contains the n-grams of the abstract. The two lists are compared to count the number of similar n-gram terms in the two lists. This work used two versions of this measure. The first uses uni-gram and the second uses bi-gram.

#### *3.2.1 Coverage measure*

The coverage of a summary was calculated automatically based on the variance of the number of sentences in the summary belongs to the three article sections: introduction, results and conclusion. To calculate this coverage measure, the sentences in the summary belonging to the introduction, result, and conclusion sections are counted

respectively followed by calculating the variance of these three numbers. If the three numbers are equal, then the variance will be zero. This means the summary covers all the sections of the article. If the variance is not zero, this means the summary focuses more on specific sections. The measure was calculated for the corresponding abstracts too to compare the coverage of the automatic summaries to the coverage of the abstracts.

Since some papers do not list sections in their abstracts, a method was devised to determine what sentences in the abstract belong to which section (Figure 7). Sentences would be taken from the abstract and the n-grams would be taken from the sentences and compared with the n-grams from the article's introduction. If the similarity score was above a threshold of 0.8, that sentence would belong to the introduction. If it was less than 0.8, the n-grams from the abstract are then compared with the n-grams from the results and if the similarity score was greater than 0.8, the sentence belongs to results. If it was less than 0.8, a final comparison was performed between the abstract n-grams and the conclusion n-grams. If the similarity score was above 0.8, the sentence belonged to the conclusion and if less than 0.8, it was discarded and not used.

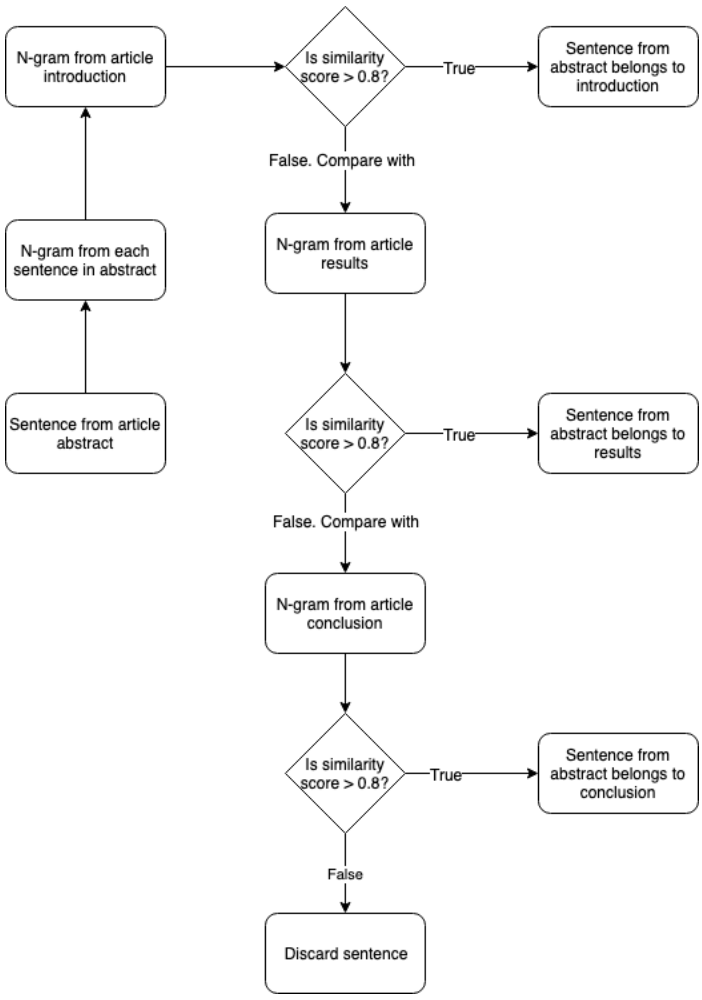



**Figure 7. Diagram flow for sentence categorization from sentences in the abstract**

The threshold of 0.8 was a determine through testing of other threshold values. First testing used a low threshold value of 0.50 and too many sentences were classified as introduction indicating too many false positives (e.g. sentences from results were classified as introduction sentences). A high threshold value of 0.95 was also tested and it was found that too many sentences were discarded with only one sentence chosen. Threshold values of 0.60, 0.65, 0.70, 0.75, 0.80, and 0.85 were also chosen and a threshold value of 0.80 showed the least amount of false positives compared to the other threshold values.

### *3.3 Testing Strategy*

The generated summary and the abstract written by the author was compared with each other using the similar-N-grams and the coverage measures to evaluate the quality of the generated summary from the perspective similarity and coverage. Each word from the summary and abstract was made into N-grams. A similarity score was generated by counting the number of items from the list of common words (words that are found both in the abstract and final summary) divided by the total number of words found in the abstract.

$$Similarity\ Score = \frac{number\ of\ items\ from\ the\ list\ of\ common\ words}{total\ number\ of\ words\ found\ in\ the\ abstract}$$

Two versions of the similarity measure were developed for comparison: the first version is based on unigram stage and the second version is based on the bi-gram stage. The unigram version looked for similarities using single grams to count how many words used in the automatic summary are used in the abstract. The bigram version looked for similarities to see if the automatic summary and the abstract are similar in sentence formation.

The same versions of the similar N-gram measure and the coverage measures were used for evaluating the performance of both of statistical-based approach and LexRank. The three hundred articles that were used in the summarization algorithm using statistical analysis were then used using the LexRank algorithm. The summarized articles from LexRank are compared with the abstract using n-grams and a similarity percentage was generated. The LexRank algorithm was implemented in Python using the existing LexRank library.

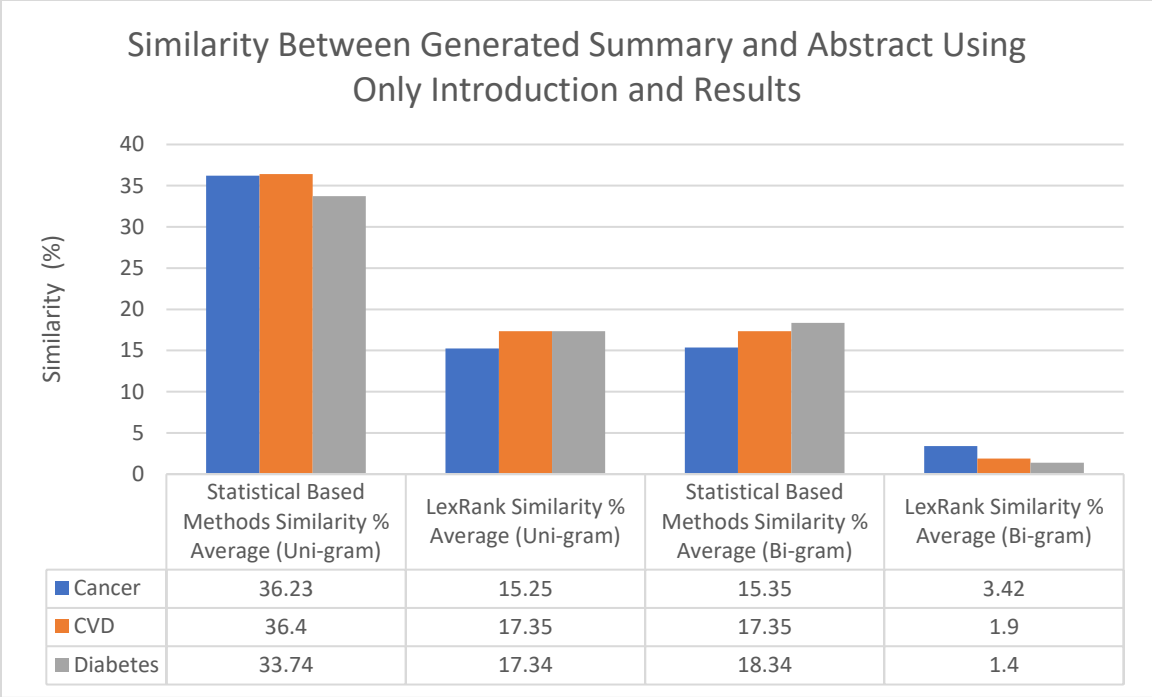
# *CHAPTER FOUR*

# *FINDINGS (ANALYSIS AND EVALUATION)*

This chapter presents the findings (analysis and evaluation) of automatically summarizing three hundred food-health articles related to cancer, CVD, and diabetes using a statistically based summarizer and machine-learning-based summarizer. Summaries were generated based on different combinations of the article sections: introduction, result and conclusion. The quality of the resulting summaries was evaluated using two measures (as described in the previous chapter). The first called Similar N-grams which measures the similarity between the generated summary and the abstract of the corresponding article. The second measure is called coverage which quantifies the coverage of the summary to the three sections of the article: introduction, results and conclusion.

## *4.1. Comparison of statistical-based and machine learning based approaches*

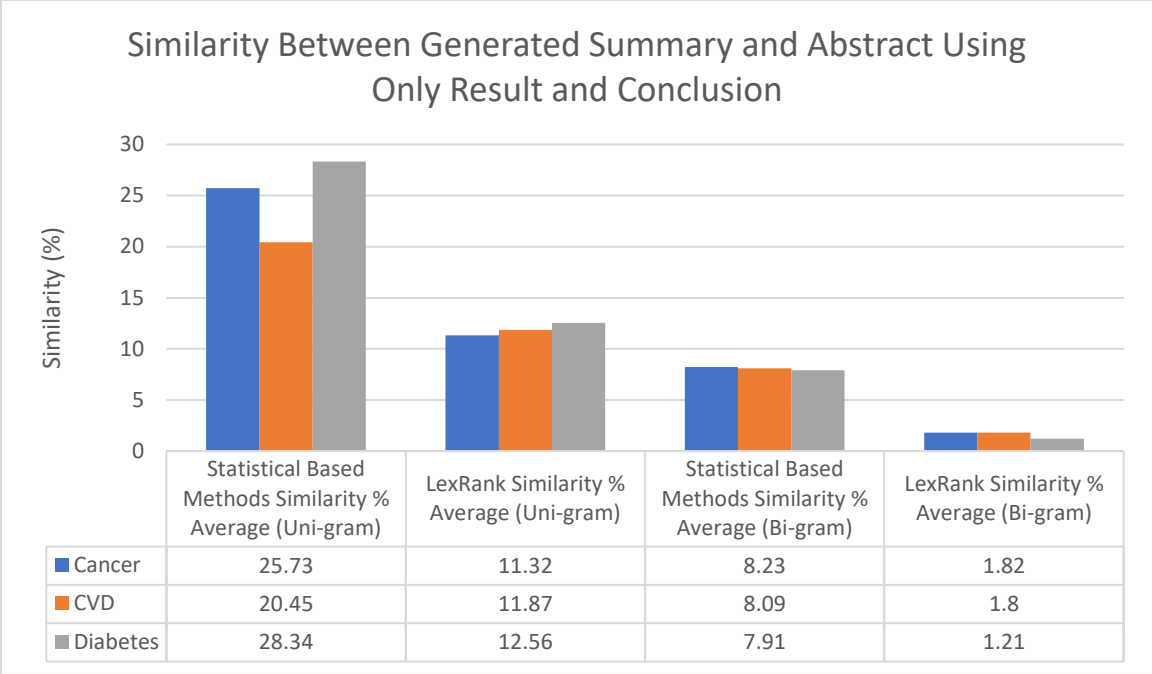
Figure 8 shows the similarity between the automatic summary generated based on three combinations of article sections using statistical-based and machine-learning-based summarization approaches. Two versions of the similarity measures were used: The first uses unigram and the second uses bi-grams.




| | Statistical Based Methods Similarity % Average (Uni-gram) | LexRank Similarity % Average (Uni-gram) | Statistical Based Methods Similarity % Average (Bi-gram) | LexRank Similarity % Average (Bi-gram) |
|---|---|---|---|---|
| Cancer | 36.23 | 15.25 | 15.35 | 3.42 |
| CVD | 36.4 | 17.35 | 17.35 | 1.9 |
| Diabetes | 33.74 | 17.34 | 18.34 | 1.4 |

**Figure 8: Percentage accuracy results from using statistical-based methods and LexRank on FHA using only introduction and results.**

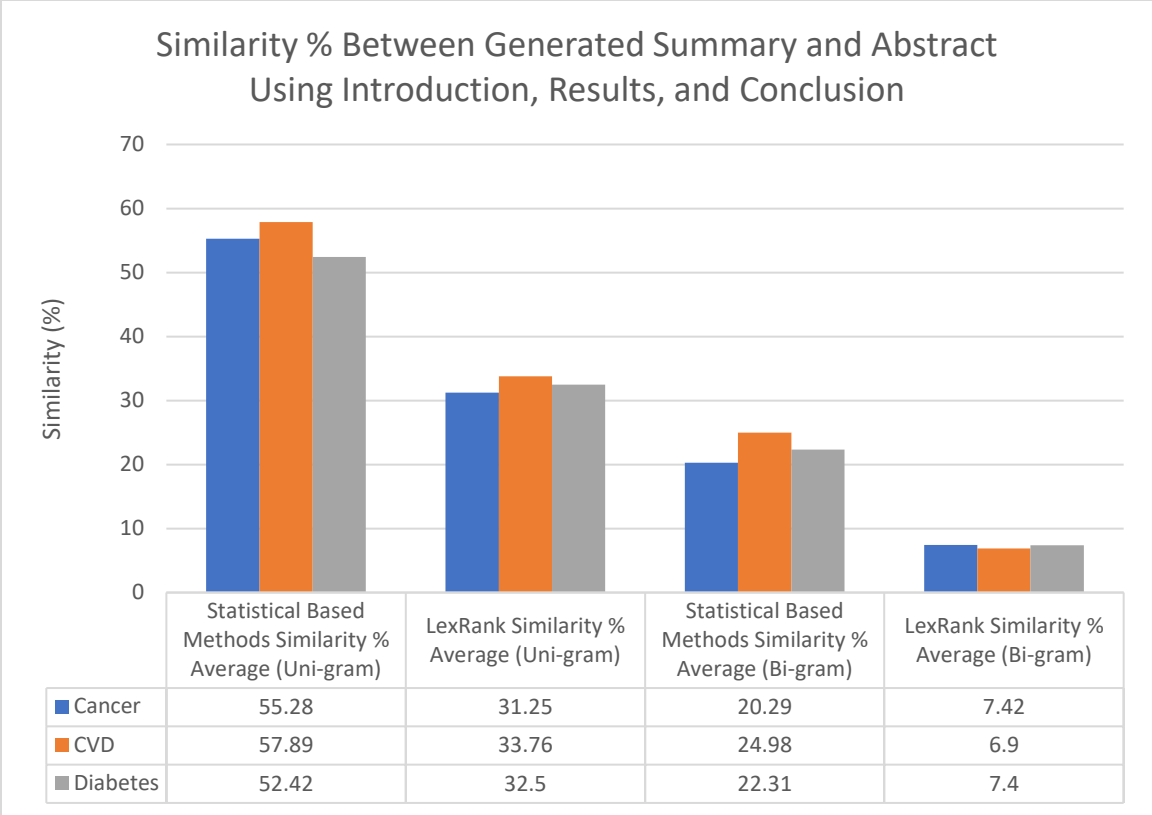
The differences between statistical based methods and LexRank differ greatly and the observed results showed that statistically based methods had a closer resemblance to the abstract than the LexRank summarizer. This difference could be attributed to the ability of the statistics to extract more information compared to the machine learning-based approaches. That relay on the ability of the approach to identify patterns.




| | Statistical Based Methods Similarity % Average (Uni-gram) | LexRank Similarity % Average (Uni-gram) | Statistical Based Methods Similarity % Average (Bi-gram) | LexRank Similarity % Average (Bi-gram) |
|---|---|---|---|---|
| Cancer | 25.73 | 11.32 | 8.23 | 1.82 |
| CVD | 20.45 | 11.87 | 8.09 | 1.8 |
| Diabetes | 28.34 | 12.56 | 7.91 | 1.21 |

**Figure 9: Percentage similarity results from using statistical-based methods and LexRank on FHA using only results and conclusion.**

Figure 9 shows the similarity of the summary and the abstract when creating a summary based on a combination of the two sections results and conclusions. The results showed that statistically based methods have a better similarity too.




| | Statistical Based Methods Similarity % Average (Uni-gram) | LexRank Similarity % Average (Uni-gram) | Statistical Based Methods Similarity % Average (Bi-gram) | LexRank Similarity % Average (Bi-gram) |
|---|---|---|---|---|
| Cancer | 55.28 | 31.25 | 20.29 | 7.42 |
| CVD | 57.89 | 33.76 | 24.98 | 6.9 |
| Diabetes | 52.42 | 32.5 | 22.31 | 7.4 |

**Figure 10: Percentage similarity results from using statistical-based methods and LexRank on FHA using introduction, results and conclusion.**

Comparing all the results (Figure 10), it is evident that having the introduction, results, and conclusion together produces the best results using both statistical based methods and using LexRank. In addition, it also shows that statistically based methods perform better than LexRank and it shows that graphical based methods are not better than statistical methods. It shows that finding words that are the most significant by word count and then using those sentences which contain the most number of significant in the sentence generate a better summary that is more similar summaries to the abstract. The results also showed a better comparison using uni-gram than bi-grams. The uni-gram from the experiment took words from the summary and abstract and made comparisons

with each other and since it was a simple comparison of whether the summary and abstract contained the same number of words. By containing the similar grams, it showed that summaries and abstract are using the same words that both consider important. Bi-grams were not a good method for comparison because since the abstract is re-written by the author and doesn't extract exact sentences from the article, the abstract is written much more differently and will not have the same words together side by side. Bi-grams produces grams that have words together side-by-side and thus the similarity percentages will be lower than uni-gram.

Comparing results from figure 8 and figure 9 demonstrates that the application of automatic summarization to the combination of introduction, result, and conclusion of FHA articles produce summaries with better similarity to the written abstracts than using only the combinations of the sections: introduction and results, or the combination of results and conclusion sections.

### *4.2. Comparing the coverage of the generated summaries and the article abstracts*

A sample of coverage calculations of nine FHA abstracts, corresponding generated summaries using statistic-based summer and summary generated using LexRank is shown in Table 1 to demonstrate how the coverage of the abstract and the summary was calculated using the introduction, results, and conclusion. We compared the coverage of the abstract and the coverage of summary for the articles of the three disease when using the two summarization approaches. The coverage is estimated based on how many sentences in the summary represent each section in the article. The variance of the number of sentences represents the coverage. The low variance indicates

that the coverage of the summary is high because its sentences are well distributed on the sections. The high variance indicates that the coverage of the summary is low because the sentences of the summary came from one section.

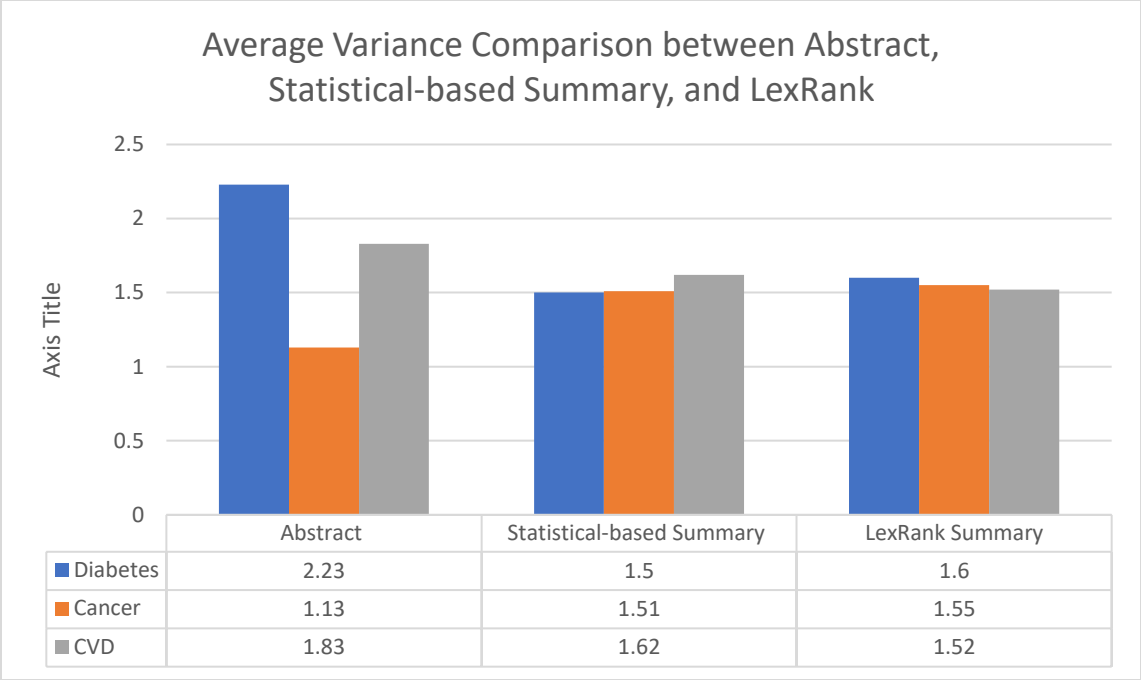
| Article title | Disease | Abstract | | | | Statistic-based summary | | | | LexRank Summary | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | How many sentences represent | | | | How many sentences represent | | | | How many sentences represent | | | |
| | | Introduction | Results | Conclusion | Variance | Introduction | Results | Conclusion | Variance | Introduction | Results | Conclusion | Variance |
| 1 | Diabetes | 1 | 6 | 1 | 8.33 | 2 | 0 | 2 | 0.33 | 3 | 0 | 1 | 2.33 |
| 2 | Diabetes | 2 | 3 | 2 | 0.33 | 2 | 1 | 1 | 0.33 | 2 | 0 | 2 | 1.33 |
| 3 | Diabetes | 2 | 4 | 2 | 1.33 | 1 | 1 | 2 | 0.33 | 3 | 1 | 0 | 2.33 |
| | Total | 5 | 13 | 5 | 21.33 | 5 | 2 | 5 | 6.33 | 8 | 1 | 3 | 13.0 |
| 4 | CVD | 2 | 4 | 1 | 2.33 | 2 | 0 | 2 | 0.33 | 3 | 0 | 1 | 2.33 |
| 5 | CVD | 2 | 2 | 1 | 0.33 | 3 | 0 | 1 | 2.33 | 3 | 0 | 1 | 2.33 |
| 6 | CVD | 1 | 4 | 1 | 3.00 | 2 | 1 | 1 | 0.33 | 3 | 0 | 1 | 2.33 |
| | Total | 5 | 10 | 3 | 13.00 | 7 | 3 | 4 | 5.33 | 9 | 0 | 3 | 21 |
| 7 | Cancer | 3 | 0 | 2 | 2.33 | 2 | 0 | 2 | 0.33 | 2 | 0 | 2 | 1.33 |
| 8 | Cancer | 2 | 1 | 1 | 0.33 | 2 | 0 | 2 | 0.33 | 2 | 0 | 2 | 1.33 |
| 9 | Cancer | 3 | 4 | 2 | 1.00 | 2 | 1 | 1 | 0.33 | 3 | 0 | 1 | 2.33 |
| | Total | 8 | 5 | 5 | 3.00 | 6 | 3 | 3 | 3.00 | 7 | 0 | 5 | 13 |

**Table 1: A comparison of the abstract and how many sentences represented each section of a scientific paper using variance**

The average variance gathering sentences from 300 article abstracts, statistic-based generated summaries, and LexRank generated summaries can be found in Table 2 and Figure 11. The average variance for the abstract was greater than the generated summaries because the number of sentences gathered for analysis in each of those sections could have greater than four and results had more sentences than the introduction and conclusion. The generated summaries collected exactly four sentences for the summary. The statistic-based summary and LexRank summary showed higher variance average in the introduction and the conclusion compared to the results which indicate that the generated summary took more sentences from the introduction and conclusion than from the results section.

| Disease | Abstract | | | | Statistic-based summary | | | | LexRank Summary | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | How many sentences represent | | | | How many sentences represent | | | | How many sentences represent | | | |
| | Introduction | Results | Conclusion | Variance | Introduction | Results | Conclusion | Variance | Introduction | Results | Conclusion | Variance |
| Diabetes | 2.4 | 4.1 | 1.8 | 2.23 | 1.43 | 1.25 | 1.31 | 1.5 | 1.33 | 1.33 | 1.35 | 1.60 |
| Cancer | 2.2 | 3.4 | 1.8 | 1.13 | 1.33 | 1.31 | 1.36 | 1.51 | 1.47 | 1.21 | 1.32 | 1.55 |
| CVD | 2.8 | 4.1 | 1.8 | 1.83 | 1.32 | 1.27 | 1.41 | 1.62 | 1.34 | 1.28 | 1.34 | 1.52 |

**Table 2: Average variance for 300 papers FHA articles (100 per disease)**




| | Abstract | Statistical-based Summary | LexRank Summary |
|---|---|---|---|
| Diabetes | 2.23 | 1.5 | 1.6 |
| Cancer | 1.13 | 1.51 | 1.55 |
| CVD | 1.83 | 1.62 | 1.52 |

**Figure 11: Comparison of average variance between the original abstract, and the generated statistic-based summary and the LexRank summary.**

The average variance between different methods of automatic comparison can be seen in Figure 11 and Table 2. The statistic-based summary and the LexRank summary also show low variance. Since there was a low sentence count (four sentences generated in each summary), only a few sentences were chosen in the summary, which may trend the variance to a low value since the sentences would not differ that much from each other. Most automatic generated summaries completely omitted the results section. This omission can be due to the fact that the results section of a paper can be quite complex depending on what kind of paper is analyzed. If a paper has a lot of numbers or statistical symbols (e.g. +/-), it can make it difficult for a summarizer to interpret that information and won't include it into the final summary.

# *CHAPTER FIVE*

# *CONCLUSION*

## *5.1 Conclusion*

Overall, this experiment showed the automatically generated summaries are not comparable to the human-made abstracts found in FHA in terms of the coverage and the similarity. Figure 6 showed the highest accuracy between the generated summary and the written abstract (52-57%). There is room for improvement by optimizing the summarization techniques' setting to specific domains. The purpose of this experiment was to show how effective different summarization techniques when summarizing FHA. The experiment found that statistically based methods performed much better than the graph-based method (LexRank) when comparing uni-grams and bi-grams. However, in terms of the overall effectiveness of automatically generated summaries, this experiment proved that they do not compare well to the author-generated abstract. However, it is not clear which one is better. the author-generated abstract could be biased while the automatic summary could ignore significant pieces of information because they were not repeated enough in the article. This requires more investigation.

## *5.2 Future Work and Improvements*

One improvement that needs to be explored is the comparison between the generated summary and the abstract. Currently, using n-grams is good for extractive summarization techniques since extractive takes sentences that already exist in the document and puts them into a summary. Comparisons can easily be made between the number of n-grams that match in the summary and the abstract – especially for uni-gram

– and works well for this experiment since this experiment used only extractive summarization. However, different methods for comparison should be explored. Summarization comparison methods for abstractive summaries should be explored and should be created because n-grams would not work abstractive summaries since these summaries create new sentences that relate to the text analyzed. These methods, if created, could theoretically work for extractive summaries as well and may provide a different way of interpreting results than n-grams.

## *REFERENCES*


Batista, J., Ferreira, R., Tomaz, H., Ferreira, R., Dueire Lins, R., Simske, S., . . . Riss, M. (2015). A quantitative and qualitative assessment of automatic text summarization systems. Paper presented at the 65-68. doi:10.1145/2682571.2797081

Bhatia, N., & Jaiswal, A. (2016). Automatic text summarization and it's methods - a review. Paper presented at the 65-72. doi:10.1109/CONFLUENCE.2016.7508049

Bui, D. D., Fiol, G. D., Hurdle, J. F.; Jonnalagadda, S. (2016). Extractive text summarization system to aid data extraction from full text in systematic review development. *Journal of Biomedical Informatics*, 64, 265-272. doi:10.1016/j.jbi.2016.10.014.

Dalal, V., & Malik, L. G. (2013). A survey of extractive and abstractive text summarization techniques. Paper presented at the 109-110. doi:10.1109/ICET

Dangeti, P. (2017). Statistics for machine learning build supervised, unsupervised, and reinforcement learning models using both Python and R. Birmingham: Packt Publishing.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research, 22, 457-479. doi:10.1613/jair.1523

Gambhir, M., & Gupta, V. (2017). Recent automatic text summarization techniques: A survey. Artificial Intelligence Review, 47(1), 1-66. doi:10.1007/s10462-016-9475-9

Goularte, F. B., Nassar, S. M., Fileto, R., & Saggion, H. (2019). A text summarization method based on fuzzy rules and applicable to automated assessment. Expert Systems with Applications, 115, 264-275. doi:10.1016/j.eswa.2018.07.047

Gupta, V., & Lehal, G. S. (2010). A Survey of Text Summarization Extractive Techniques. *Journal of Emerging Technologies in Web Intelligence*,*2*(3). doi:10.4304/jetwi.2.3.258-268

Hassler, M., & Fliedl, G. (2006). Text preparation through extended tokenization. Data Mining VII: Data, Text and Web Mining and Their Business Applications. doi:10.2495/data060021

Hu, Y., Chen, Y., & Chou, H. (2017). Opinion mining from online hotel reviews – A text summarization approach. Information Processing and Management, 53(2), 436-449. doi:10.1016/j.ipm.2016.12.002

Jonathan J. Webster and Chunyu Kit. 1992. Tokenization as the initial phase in NLP. In Proceedings of the 14th conference on Computational linguistics - Volume 4 (COLING '92), Vol. 4. Association for Computational Linguistics, Stroudsburg, PA, USA, 1106-1110. DOI: https://doi.org/10.3115/992424.992434

Kaynar, O., Gormez, Y., Isik, Y. E., & Demirkoparan, F. (2017). Comparison of graph based document summarization method. Paper presented at the 598-603. doi:10.1109/UBMK.2017.8093475

Leiva, L. A. (2018). Responsive text summarization. Information Processing Letters, 130, 52-57. doi:10.1016/j.ipl.2017.10.007

Liang, X., Qu, Y., & Ma, G. (2012). Research on extension LexRank in summarization for opinionated texts. Paper presented at the 517-522. doi:10.1109/PDCAT.2012.117

Lloret, E., Romá-Ferri, M. T., & Palomar, M. (2013). COMPENDIUM: A text summarization system for generating abstracts of research papers. Data & Knowledge Engineering, 88, 164-175. doi:10.1016/j.datak.2013.08.005

Liu, W., Luo, X., Zhang, J., Xue, R., & Xu, R. Y. D. (2017). Semantic summary automatic generation in news event. Concurrency and Computation: Practice and Experience, 29(24), e4287-n/a. doi:10.1002/cpe.4287

Moratanch, N., & Chitrakala, S (2017). A Survey on Extractive Text Summarization. *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP).* doi: 10.1109/ICCCSP.2017.7944061

Nandhini, K., & Balasundaram, S. R. (2014). Extracting easy to understand summary using differential evolution algorithm. Swarm and Evolutionary Computation, 16, 19-27. doi:10.1016/j.swevo.2013.12.004

Raphal, N., Duwarah, H., & Daniel, P. (2018). Survey on abstractive text summarization. Paper presented at the 0513-0517. doi:10.1109/ICCSP.2018.8524532

Report of Canada's Economic Strategy Tables: Agri-food - Economic Strategy Tables. (2018, September, 28) Retrieved December 10, 2018, from https://www.ic.gc.ca/eic/site/098.nsf/eng/00022.html

Ross, M., Mahmoud, E. S., and Abdel-Aal, E. M. (2018). Exploring Identifiers of Research Articles Related to Food and Disease using Artificial Intelligence,

International Journal of Advanced Computer Science and Applications (IJACSA), 9(11).

Varalakshmi K, P. N., & Kallimani, J. S. (2018). Survey on extractive text summarization methods with multi-document datasets. Paper presented at the 2113-2119. doi:10.1109/ICACCI.2018.8554768

Yang, G., Wen, D., Kinshuk, Chen, N., & Sutinen, E. (2015). A novel contextual topic model for multi-document summarization. Expert Systems with Applications, 42(3), 1340-1352. doi:10.1016/j.eswa.2014.09.015

Yang, K., Al-Sabahi, K., Xiang, Y., & Zhang, Z. (2018). An integrated graph model for document summarization. Information, 9(9), 232. doi:10.3390/info9090232

Zhong, S., Liu, Y., Li, B., & Long, J. (2015). Query-oriented unsupervised multi-document summarization via deep learning model. Expert Systems with Applications, 42(21), 8146-8155. doi:10.1016/j.eswa.2015.05.034