**SumMe:**

**Filipino-English Summarizer using an abstractive semantic-based approach**

A Thesis

Presented to the Faculty of the

College of Computer and Information Sciences

Polytechnic University of the Philippines

In Partial Fulfillment

Of the Requirements for the Degree

Bachelor of Science in Computer Science

Aranzamendez, Paul Justin A.

Capalad, Justin Allen

Evangelista, Renz Gio A.

March 2015

**AUTHORIZATION**

We hereby declare that we are the sole author/s of the thesis.

We authorize the Polytechnic University of the Philippines and the College of Computer and Information Sciences to lend this thesis to other institutions or individuals for the purpose of scholarly research.

We further authorize the Polytechnic University of the Philippines and the College of Computer and Information Sciences to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Aranzamendez, Paul Justin A.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Capalad, Justin Allen

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Evangelista, Renz Gio A.

March 2015

The thesis **“SumMe: Filipino-English Summarizer using an abstractive semantic-based approach”** submitted and presented by Paul Justin A. Aranzamendez, Justin Allen Capalad, Renz Gio A. Evangelista in partial fulfilment of the requirements for the degree of Bachelor of Science in Computer Science has been

Recommended for Approval and Acceptance: Date:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Faculty-In-charge

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ria Ambrocio Sagum, MCS

Thesis/Technical Advisor

Approved by the Thesis Defense Panel:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Ria Sagum

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Carlo Inovero

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Michael dela Fuente

Accepted for the Department of Computer Science

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Michael dela Fuente, MSGITS

Department Chairperson

Accepted for the College of Computer Management and Information Technology

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Gisela May A. Albano

Dean of the College

**ABSTRACT**

Automatic Summarization is a research area that can be applied to a variety of domains. Summarization can be either extractive, which is basically a subset of the original document, or abstractive, which employs deeper understanding of the source document and utilizing natural language generation to truly create a summary “from one’s own words” and not just an extract of the source.   
One of the most common domain in automatic summarization are news articles, with researchers coming up with novel ideas of summarization for different purposes, employing different summarization types and kinds of summarization methods.

However, one particular thing to be noticed in this area is that most summarization systems have little to no tolerance to code-switching texts. Code-switching texts are texts that have a mixture of two languages in a sentence or paragraph that might confuse a reader that is not particularly knowledgeable on one of the languages in the code-switch. This scant tolerance in automatic summarization might result to a lower coherence and cohesion in the resultant summary.

In this paper the researchers will tackle Single document summarization using an abstraction method based on semantics. Aside from that, we also keep in mind the possibility of a code-switch in the contents of the text, specifically Filipino-English. The study is experimental and the researchers will use experimental methodologies to measure the effectiveness and accuracy of the system.

**Table of Contents**

[CHAPTER 1 1](#_Toc412687625)

[1.1 Introduction 1](#_Toc412687626)

[1.2 Background of the Study 3](#_Toc412687627)

[1.3 Statement of the Problem 3](#_Toc412687628)

[1.4 Research Hypothesis 4](#_Toc412687629)

[1.5 Conceptual Framework 4](#_Toc412687630)

[1.5.1 Conceptual Framework of the System 4](#_Toc412687631)

[1.5.2 Conceptual Framework of the Study 5](#_Toc412687632)

[1.6 Significance of the Problem 6](#_Toc412687633)

[1.7 Scope and Limitations 7](#_Toc412687634)

[1.7.1 Scope of the Study 7](#_Toc412687635)

[1.7.2 Limitations of the Study 7](#_Toc412687636)

[1.7.3 Scope of the System 7](#_Toc412687637)

[1.7.4 Limitations of the System 7](#_Toc412687638)

[1.8 Definition of Terms 7](#_Toc412687639)

[CHAPTER 2 10](#_Toc412687640)

[2.1 Review of Related Study 10](#_Toc412687641)

[CHAPTER 3 25](#_Toc412687642)

[3.1 Methodology 25](#_Toc412687643)

[3.2 System Architecture 26](#_Toc412687644)

[3.3 Research Paradigm 28](#_Toc412687645)

[3.4 Population of the Study 29](#_Toc412687646)

[3.5 Types of Sampling Technique 29](#_Toc412687647)

[3.6 Data Gathering Procedure 29](#_Toc412687648)

[3.6.1 Preliminaries 29](#_Toc412687649)

[3.6.2 Experimental Method 30](#_Toc412687650)

[3.7 Research Instruments 31](#_Toc412687651)

[3.8 Statistical Treatment 31](#_Toc412687652)

[CHAPTER 4 34](#_Toc412687653)

[CHAPTER 5 37](#_Toc412687654)

[5.1 Summary 37](#_Toc412687655)

[5.2 Conclusions 38](#_Toc412687656)

[5.3 Recommendations 39](#_Toc412687657)

[Bibliography 40](#_Toc412687658)

[Appendix A 42](#_Toc412687659)

[Graphical User Interface 42](#_Toc412687660)

[Appendix B 47](#_Toc412687661)

[Implementation Report 47](#_Toc412687662)

[Appendix C 50](#_Toc412687663)

[Experiment Paper 50](#_Toc412687664)

[Appendix D 53](#_Toc412687665)

[Letter for Expert 53](#_Toc412687666)

[Appendix E 55](#_Toc412687667)

[Photos of Implementation with Expert 55](#_Toc412687668)

[Appendix F 57](#_Toc412687669)

[Computation of the Results of the Study 57](#_Toc412687670)

**List of Figures**

[Figure 1 IPO Illustration of the System 5](#_Toc412687072)

[Figure 2 IPO Illustration of the Study 6](#_Toc412687073)

[Figure 3 System Architecture 27](#_Toc412687074)

[Figure 4 Iterative and Incremental Development 28](#_Toc412687075)

**List of Tables**

[Table 1 Table of Compression Ratio 34](#_Toc412687221)

[Table 2 Table of Retention Ratio 35](#_Toc412687222)

[Table 3 Table of Q&A Task Performance 35](#_Toc412687223)

[Table 4 Summary of Results Measured 36](#_Toc412687224)

# CHAPTER 1

**THE PROBLEM AND ITS BACKGROUND**

This chapter shows the Introduction of the study, its background, the scope and limitations of the study. This chapter states problem that the study needs to answer and improve.

## 1.1 Introduction

A news article discusses current or recent news of either general interest (i.e. daily newspapers) or of a specific topic (i.e. political or trade news magazines, club newsletters, or technology news websites). A news article can include accounts of eyewitnesses to the happening event. It can contain photographs, accounts, statistics, graphs, recollections, interviews, polls, debates on the topic, etc. Headlines can be used to focus the reader’s attention on a particular (or main) part of the article. The writer can also give facts and detailed information following answers to general questions like who, what, when, where, why and how. (Article (Publishing)).

Summarization is the art of abstracting key content from one or more information sources (Hahn, 2001), and automatic summarization is the process of using a computer program to create a summary that retains the most important parts of the original document (Automatic Summarization).

News nowadays are very detailed and narrates happenings and events vividly. News has been an important part in people's lives, most people starts the day reading or watching news. It helped people to know what's happening today and when someone ask about the news the person would just say the same thing a line of a song said, "*How do I know? I read it in the 'Daily News*" (Paxton).

People reads the Header of every news on Newspapers or on-line to get a thoughtful idea of what the news contains, problems are to wordy articles, and some are redundant that contains looping statements and information. Some writers tends to write long articles because of the Pay-By-The-Word system which the pay a writer gets depends on how many words that writer has written, but this system sometimes leads to redundant news. A blog by a writer once stated that "*Writers are terrible when it comes to talking about money. Many of them feel guilty or are afraid to negotiate for more money, which is insane, but true...Pay-by-the-word discourages reporting, does not reward more talented writers and adds an invisible siphon to editorial budgets.*" (Donatelli, 2013).

Information that news holds are important even if the news was way back 5 years ago. A song by The Rolling Stones said: "*Who wants yesterdays papers?, living a life of constant change, every day means the turn of a page*" (Stones, 1967).There are those researchers and people interested on some cases may read back news, but the idea is to provide summary of only news from before for people that are looking for specific news to read the whole only if that person found it interesting by reading the summary. Summary is shortening body of text and getting only the important details in it.

A summary on is very important just like some flashback news on the TV it holds small portion of time to be read or watched, but it has all important details there is on news. An episode of the TV series Breaking Bad titled "Face Off" shows the news the characters watched states brief but informative fraction of the whole news, where the Main characters are watching the news about an explosion that killed 3 people. The study shows that you can read news on-line without actually reading the whole part of it. Getting only the important details of the news helps lessen the time of scanning stacked news on-line.

## 1.2 Background of the Study

Automatic Text Summarization is a process where a computer summarizes text and returns the text with shorter and with less redundancy. Automatic text summarization has two approaches, abstractive and extractive. Extractive summarization is identifying the most relevant passages in one or more documents, the passages which are sentences or phrases are extracted and pasted together to form shorter text compared to the original with little loss of information. Abstractive summarization compared to extractive is more difficult and challenging it parses the original text in a deep linguistic way and interprets the text in a semantical representation and then generates shorter text, abstract that has the same informational content (Hassel M. , Evaluation of Automatic Text Summarization, 2004). An abstractive approach requires a representation of the text that serves as an intermediate step before the generation of new sentences. (Etienne Genest & Lapalme)Two properties must be measured in evaluating summarized text, the Compression Ratio (how short the summary compared with the original) and Retention Ratio (how much information retained) (Dalianis & Hassel). It is often important to know where the subject learned the information reflected in the summary. In research on the subject's reasoning after reading multiple documents, it is similarly important to know which documents have the most influence on the subject's recall. (Foltz, 1996)

## 1.3 Statement of the Problem

The study aims to summarize news articles written in Filipino and English using abstraction methods under Semantic-based summarization.

**1.**  What is the rating of the system-generated summaries by SumMe in terms of:

a. Compression ratio

b. Retention Ratio

c. Understandability/Coherence

d. Cohesion

e. Q&A Task performance

## 1.4 Research Hypothesis

There is no significant difference between the evaluation of the SumMe and TextTeaser systems mentioned in terms of.

## 1.5 Conceptual Framework

### 1.5.1 Conceptual Framework of the System

The system will take as input a News article, either in a text file or text on the text area. The taken input will undergo three major processes which are the Preprocessing, the Semantic processing, and the Natural Language Generation. Preprocessing prepares the unstructured text into data that is readable by the Semantic processing block. Here, the data is processed to be a semantic representation which is then reduced and interpreted. The resulting interpretation is then fed into the Natural Language Generation block which then generates sentences and paragraphs for the given interpretation, which is the summary of the given input. **Fig 1** shows the Conceptual Framework of the system.

\* Input Text file/news article either Filipino or English

\* Pasted news article that will be summarized

\* Input contents undergo preprocessing, semantic representation, reduction and interpretation, which is then fed into the Natural Language Generator

\* Summarized input news article

**INPUT**

**PROCESS**

**OUTPUT**

**Figure 1 IPO Illustration of the System**

### 1.5.2 Conceptual Framework of the Study

The Study starts on gathering News articles taken from selected Online Resources, the gathered news articles will be used as input on the System. The output summarize news article of the system will be compared to the input news article as to measure the Retention Ratio by giving expert(s) Experiment papers and Coherence and Cohesion will be measured using the Web system Coh-Metrix. The Compression Ratio will be measured by the developers as it only includes counting sentences numbers. On the experiment paper expert(s) are required to hypothetically formulate questions from the information given on the input article and by answering the questions using information from the output the Answer Recall Lenient (ARL), Answer Recall Strict (ARS) can be measured to get the Answer Recall Average (ARA).**Fig 2** shows the Conceptual Framework of the study.

\* Recent news articles from selected websites (Filipino and English)

\* Output Summary of the System

\* Experimental Method

\* Expert(s) will compare the input and output of the system for Retention Ratio computation

\* Expert(s) are ask to make questions from the input article that must be answered by the output

\* List of SVO candidates extracted from the input and weighted, Subject, Verb and Objects are scored. The system will get the 10 highest scored

**INPUT**

**PROCESS**

**OUTPUT**

**Figure 2 IPO Illustration of the Study**

## 1.6 Significance of the Problem

This study might benefit:

* Internet users, as the system will give them a gist of the news article first, giving minimal effort for readers to determine if a certain news article is of use to their intent.
* Students, News readers and secretaries might find this useful as it would make easier for them to find articles that match their interest, since the system’s output will show the general idea of the input article.
* The study might also help Researchers as the study is an Abstractive summarization with graphical representation and the first one in English and Filipino.

## 1.7 Scope and Limitations

### 1.7.1 Scope of the Study

The study will focus on the development and the assessment of the performance of the application with regards to of the summary of news articles. The study will show if there is a significant difference between the proposed system and other existing news summarization systems.

### 1.7.2 Limitations of the Study

The study will only span the assessment of the proposed application and a select number of existing news summarization systems in terms of evaluation metrics which are: Compression ratio, Retention Ratio, Understandability/Coherence, Cohesion and Q&A Task performance. The sample news articles will only be taken from rappler.com, inquirer.net, and the Philippine star.

### 1.7.3 Scope of the System

The system will summarize news articles from an input text file or pasted text. The system will parse both raw text and html encoded documents. The content will use preprocessing techniques and Semantic-based abstraction approach in summarizing the given document. The system will be a web app for easy access in the web.

### 1.7.4 Limitations of the System

The System will only accept input that is either in Filipino, English or both. The corpus that will be used for the Filipino language is comprised mostly of Tagalog words, with no regard for the other dialects.

## 1.8 Definition of Terms

**Abstraction-** is the forming of ideas from specific given samples.

**Abstractive Summarization**- parses the original text in a deep linguistic way and interprets the text in a semantical representation and then generates shorter text that is not redundant from its original form.

**Automatic Summarization -** is the process where a system will summarize the article or news given by the user. The output will only contain the most important words from the input.

**Automatic Text Summarization -** is a process where computer summarizes text and lessen redundancy, but keeps the coherency. Automatic Text Summarization is based on statistical, linguistically and heuristic methods. A summarization system calculates the occurrences of keywords on texts to give it frequency, the higher the frequency the more important a keyword is on a sentence.

**Compression Ratio -** measures the shortness of summary of the output over the original length of the source document.

**Lexicon -** is a dictionary where semantically generated text will be stored to form structured texts. Lexemes are words arranged in an alphabetical order of a specific language, it contains tags of part of speech for every lexeme. Lexemes also contain atomic and bound morphemes that cannot stand alone for it only contains words that have no tenses or forms. A central role of the lexicon is the documenting of established lexical norms and conventions.

**Natural Language Generation –** aims to write texts with the same quality as human it deals with rule-bound text generation. NLG starts from data to result syntactically correct sentences that that has been interpreted and analyze, it is used to provide coherent and concise output. NLG structures words on their morphological forms can explain arguments

**Natural Language Processing -** is a branch of Artificial Intelligence that deals with human language (natural language) to deal with mechanical problems either spoken or written. It deals with the application of computational models to text or speech data.NLP involves fundamental questions of how to structure formal models of natural language phenomena, and of how to design algorithms that implement these models. NLP connects people and technology through electronic representation of human languages.

**News Article -**is a body of text that contains news entities of current information, researches and events that happened on daily bases. this will serve as the input of the system

**Parts of Speech Tagger -** will serve as an indicator or a trace for each word in a sentence. Each word in the sentence will have their tag according to the part of speech they are categorized.

**Question Game -** is a method that will be used to measure the accuracy of the expected output, it involves proponents that will read the output on three ways: the output without reading the original text, the output after reading the source document and reading the source document.

**Retention Ratio -** measures the output summarized text on how much information will be retained over the information on the source document.

**Semantic Graph -** the rules and standards for the correct structures of a sentence.

**Semantic Ontology -** serves as a database for the proper and coherent sentence.

**Sentence Splitter -** is a pre-process that will split every sentence on each paragraph of the given news and article

**Summary-** will be the output after the application is finished on processing and evaluating the given input

# CHAPTER 2

**REVIEW OF RELATED LITERATURES AND STUDIES**

This chapter covers the different literatures and studies that benefit and will be used as guidelines throughout the whole research. This chapter shows overview of the gathered literatures and studies from different researchers and authors that show relevance to the study and how it will affect the research and the study made by the researchers of this study.

## 2.1 Review of Related Study

Automatic Text Summarization is split into two groups, text extraction and text abstraction. The text extraction identifies the most relevant passages in one or more documents. These passages, often sentences or phrases being extracted then combine together to form a non-redundant summary that is shorter than the original document (Hassel M. , Evaluation of Automatic Text Summarization, 2004).

Text abstraction is harder to do than text extraction in a way that it needs to parse the original text and interpreting the text semantically to produce a formal representation (Hassel M. , Evaluation of Automatic Text Summarization, 2004). The product of the text abstraction is much closer to what humans might produce when summarizing a document.

One of the important elements of summarization is to distinguish the part of speech of each word in a sentence. The significant duty in summarization is identifying the topic by finding the cue phrases.

To enhance the text summarizer and to acquire the topic of the sentence, Martin Hasselused Name Entity Recognition. The job of the Named Entity Recognition is to classify proper nouns the document. Clearly, the Name Entity carries clues to the topic of the document. But one of the problems they encountered in NER is a serious losses of sentences by prioritizing the elaborative sentences over introductory (Hassel M. ). Another problem is repetitive of the summary for the reason that a certain word is being repeated for numerous times in a document.

One of the existing summarization nowadays is the SweSum summarization engine (Dalianis & Hassel).SweSum is the first automatic text summarizer for Swedish news text created by Dalianis and Hassel. SweSum uses the most common text extractor paradigm. The method of this paradigm is to extract the most significant phrase or sentences from a text to create a new shorter non redundant text (Hassel M. , Evaluation of Automatic Text Summarization, 2004).

The domain of the SweSum is Swedish newspaper text. It utilizes several different topic identification schemes. One of the schemes is the bold tag used to emphasize contents of the text. Headings are given a higher weight. The most relevant information is always presented at the beginning of the newspaper text and are given higher scores. According to the others who used the engine that the performance is estimated to be as good as the state of the art techniques for English. Good Summaries at compression rates around 70% (retaining 30% of the words) can be obtained for original text of two to three pages in the news domain (Dalianis & Hassel, 2001). To identify the important topics of a text, Named Entity Recognition can enhance the identification of key text segments (Hassel M. ).

Additionally, the preprocessor for Swesum is SweNam that tags all found Named Entities with one of the four possible categories – name of persons, locations, companies, brands, products and time stamp. According to Dalianis &Astrom, the Named Entities found by SweNum are fairly reliable with a 92 percent precision. The recall is only 46 percent (Hassel M. ).

One of the problem of the using the Name Entity module tends to prioritize elaborative sentences over introductory that is responsible for some losses of background information. However, Name Entities clearly carry clues to distinguish the topic of a text. It is also helpful in knowing the main part of the different participants and their respective role in a text (Hassel M. ).

The SweSum works in three different passes, the first one is the tokenization and keyword extraction. The second is the ranking of the sentences and the last pass is the produced summary (Delianis & Hassel, How Short is Good? An evaluation of automatic summarization). SweSum’s domain consists mainly of Swedish HTML tagged newspaper text. SweSum ignores HTML tags that control format of the page but processes the HTML tags that control the format of text (Delianis & Hassel, How Short is Good? An evaluation of automatic summarization).Also the web page allows the user to specify the text to be summarized, and the degree of summarization. Besides that, the user is asked what language to be used for the text to apply what language-specific resources (Delianis & Hassel, How Short is Good? An evaluation of automatic summarization).

Furthermore, the function of the tokenization is to split the text into sentences. The SweSum then execute topic detection and identifies the important parts of the text by assigning scores to sentences according to the criteria. The summarizer uses dictionary with about 700.000 that contains the open class words and their stems (Delianis & Hassel, How Short is Good? An evaluation of automatic summarization).

Another automatic text summarizer is the GreekSum, it is based on well-known SweSum summarization engine. Basically it is a text summarizer for Greek news that uses a Greek key word dictionary provided by NCSR Demokritos, Athens (Pachantouris, 2004-2005).

Greek native speakers used SweSum’s built in function of Generic Summarization when evaluating the GreekSum [NewsByte].

A further method that is implemented by Martin Hassel is using a text pre-processor, called PRM (Pronoun Resolution Module) written in Perl and works in SweSum. (Hassel M. , Pronominal Resolution in Automatic Text Summarization, 2000) PRM uses lists of likely focuses (Sidner, 1984) and uses lexicon of nouns that contains information about each entry’s natural or grammatical gender (Hassel M. , Pronominal Resolution in Automatic Text Summarization, 2000). The procedure of this is that the nominal phrase is recognized and categorized then move it to a suitable list of category. They also found that using the PRM in the SweSum greatly improve the coherence for text rich with pronouns.

In the paper of Yihong Gong and Xin Liu, they propose two generic text summarization methods that create text summaries by ranking and extracting sentences from the original documents. The first method is IR methods that ranks sentences significance, while the other method latent semantic analysis technique that identifies important sentences semantically (Gong & Liu). They also compared the manual summarization created by human to the summaries of the generated by the two methods mentioned. Their observation on the experiment shows the larger the documents are, the more disparities between the summaries of humans and the summarization generated by the text summarization.

A system from SRA international, Inc. named The Knowledge Management (KM) uses extraction method for summarization. It features using morphological analysis, name tagging and co-reference resolution. To determine the optimal combination of the features in combination with statistical combination from the corpus to identify the best sentences to include in a summary, they used a machine learning technique (sra.com).

Another research made by Dalianis et al. is the ScandSum that aims to for the development on summarization tools particularly for the Scandinavian languages (Danish, Norwegian and Swedish). ScandSum is also uses extractive text paradigm and based on the SweSum. The research has been successfully ported Danish to Norwegian (Delianis & Hassel, How Short is Good? An evaluation of automatic summarization).

NorSum is the same as the SwesSum with the cooperation with the ScandSum network. They decided to collect a corpus containing manual summaries from the Norwegian newspaper, Bergens Tidende. The editorial works in the newspaper articles were shortened by removing the last few sentences or rewriting the whole text. Before inserting the news, it was slightly edited to fit the right format and were automatically divided into sentences that were each given a unique ID.

The OGI/OHSU baseline multilingual multi-document summarization system is a summarization system developed by Fisher et al (2000).

They use sentence ranking based on simple features and sentence selection from a ranked list. SVMlight is used to learn preference ranking and use English text sentences when selecting from the ranked list.

According to Angheluta et al. (2002) study, they have developed a topic segmentation algorithm. It detects the structures in the text by using the generic text structure cues. They called the process “Layered topic segmentation that correlate key terms with each topic or subtopic and outputs a tree-like table of content (TOC). The trees use the most significant terms at general and more specific levels of topicality.

The result of the TOCs makes both single-document abstracts and the multi-document abstracts and extracts.

Additionally, there are divisions in methods for evaluation automatic text summarization system, namely into intrinsic and extrinsic evaluation methods (Hassel M. ).

The intrinsic evaluation measures the system in of itself. It is done by comparing gold standard, made by a reference summarization system. The main focus of the intrinsic evaluation is on the coherence and how informative the summaries are (Hassel M. ).

While the extrinsic evaluation measures the efficiency and acceptability of the generated summaries like the relevance assessment or reading comprehension (Hassel M. ).

Extraction-based summarization sometimes suffers from coherence problem like anaphors or gaps in rhetorical structure of the summary.

Moreover, the precision and recall figures can be used to assess the performance in utility figures and content based methods. Precision and recall are often combined in F-Score for standard measures. One of the problem of this measure is that they are not efficient in distinguishing between many possible and the summaries differ in the content (Hassel M. ).

The KTH extract Corpus tool assists in the collection of extract-based summaries provided by human informants and semi-automatic evaluation of machine generated extracts in order to easily evaluate the SweSum summarizer. The corpus contains a number of original full texts and several man-made extracts for each text (Delianis & Hassel, Generation of Reference Summaries, 2005).

The compiled of the extracted corpus can analyzed automatically that the inclusion of the sentences in the several extracts for a given source text can easily be compared. The development of an automatic summarizer allows a quick adjustment and evaluation cycle (Delianis & Hassel, Generation of Reference Summaries, 2005).

In addition, the KTH extract tool collects the statistics on how many times a specific extract unit from a text has been included in a number of different summaries. The model summary can be composed using only the most frequently chosen sentences (Delianis & Hassel, Generation of Reference Summaries, 2005).

The group of text from two Swedish and one Danish group have been collected during iterations.

The Swedish extract corpus consists of a total of 301 Swedish text extracts submitted by 45 informants; average length of submitted extracts is currently 32.5 percent (31% and 34% for group 1 and group 2) (Delianis & Hassel, Generation of Reference Summaries, 2005).

While in Danish extract corpus consists of 135 Danish text extracts submitted by 15 informants; average length of submitted extracts is currently 32%.

Numerous equally good summaries for one specific source text effectively making the evaluation against one rigid reference summary unsatisfactory in the automatic text summarization. The evaluation methods have different compression rates that are optimal for different text types or genre, or even within a different text type or genre (Delianis & Hassel, Generation of Reference Summaries, 2005).

The research on summary generation technique mostly relies on the extraction of salient sentence from the original to generate the summary. Several methods for determining the importance of a sentence has been developed. Some algorithms calculate scores for each sentence considering the location of the sentence and word frequencies (Dalianis et al. 2003), while others use semantic information (Wordnet, for example), in order to find the hierarchy of concepts.

Natural Language Generation, the study of human language generation, is a multidisciplinary enterprise, requires expertise in areas of computer science and linguistics. The aim is to learn how computer program produce high-level, narural language from computer depiction of information (Batang, 2006).

Furthermore, according to Herzog (1999), natural language generation often is characterized as a process that has to start from the communicative goals of the writer or speaker and needs to employ some sort of planning to progressively convert them into written or spoken words. With this in mind, the goal of the language producer is force into linguistic in nature, trying to produce particular words. The generation of the summary uses two techniques to generate a summary; strategically (deciding what to say) and tactically (deciding how to say it). Determining the large-scale structure of the text to be generated should also include content selection. Usually, this process involves a tree-like formation in which the leaves contain instructions that are then passed in turn to a sentence generator, task that can be further categorized into sentence planning (Batang, 2006).

There are different types of generation can be classified into four main categories:

The first one is the Canned text systems that uses the simplest approach for single-text sentence and multi-sentence text generation. They are trivial to create, but very inflexible (Batang, 2006).

The second one is the Template systems that are the next level of complexity. It relies on the application of pre-defined templates or schemas and is able to support flexible alterations. The template approach is used mainly for multi-sentence generation, particularly in applications whose text is fairly regular in sentence (Batang, 2006).

The third one is the Phrase-based systems that employ what can be seen as generalized templates. In such systems, a phrasal pattern is first selected to match the top level of the input, and each part of the pattern is recursively expanded into a more specific phrasal pattern that matches some sub portion of the input. At the sentence level, the phrases resemble phrase structure grammar rules and at the discourse level, they play the role of text plans. The last one is the Feature-based systems, which are as yet restricted to single-sentence generations, represent each possible minimal alternative of expression by a single feature. Accordingly, each sentence is specified by a unique set of features. In this framework, generation consists in the incremental collection of features appropriate for each portion of the input. Feature collection itself can be based either on unification or on the traversal of a feature selection network. The expressive power of the approach is very high since any distinction in language can be added to the system as a feature. Sophisticated feature-based generators, however, require very complex input and make it difficult to maintain feature interrelationships and control feature selection (Batang, 2006).

Many natural language generation systems follows a hybrid approach by combining components that utilize different techniques. This study used the summarization by abstraction with the Phrase-Merging summary generation (Batang, 2006).

The online News Summarizer by Cruz et al. (2003) is a direct application of an automatic text summarization system. It is an email service, which seeks to provide its users with daily, summarizes from various local and international news entities. Its primary component is an auto-text summarization, which uses SVM machines. Instead of using traditional approach to automate text summarization such as lexical chains, machine learning based summarization, discourse trees, and sentence extraction techniques; the researchers used vector machines SVMs in determining which sentences are possibly included in the summary. The main component of the project is the summarizer engine. It first takes as input text file of the news article. It segments the text into words and sentences. It then scores these words based on a certain number of features. The scores are then input for the SVM module, which determines which sentences, are important, and should be included in the summary. The summarizer then produces the cohesive summary. The system will produce summarization results at three levels: keywords, salient sentences, rough summaries – this is the order of increasing complexity and increasing difficulty of their automatic generation. They used method in their summarization. In preprocessing method, it consists of lemmatization (finding root forms) and tagging (finding part of speech classes), and will use lexical database WordNet (Miller, 1990). The system uses syntactic knowledge and frequency analysis of the text to classify noun phrases into five categories of technicality. It also performs automatic syntactic category disambiguation based on text statistics, using the categories of the neighboring words (Batang, 2006) (Philippines, 2002)

In Determining Keywords method, the Online News Summarizer identified keywords using two methods. First, by considering the frequency of candidate noun groups in a large corpus, and second, by giving a large number of candidate rules to an inductive learning system. Examples of such rules that may help identify keywords are “select noun-phrases that occur frequently in the first paragraph” or “select phrases that occur in titles of sections.” Next, the system is trained on a body of summarized texts, and only the rules that perform well trained (Batang, 2006) (Philippines, 2002).

In Determining Salient Sentences method, Salient sentences will be determined by first identifying the activities and objects relevant to the subject matter communicated by the text. These will be essentially keywords, determined in the Determining Keywords phase. The sentences that convey the most information about the subject matter will then be selected. Selection will be based on a relevancy measure, obtained by counting referential links between the salient activities and objects (Batang, 2006) (Philippines, 2002).

In Assembling Summaries method, they applied sentence truncation. A sentence can be truncated in an orderly way by applying one of predefined discourse-level heuristics, triggered by the presence of specific lexical elements. For example, the sentence “the use of large, public domain linguistic resources,i.e.text corpora and on-line lexicons” can be compressed to “the use of large, public domain linguistic resources.” The relevant heuristic suggests dropping the part that follows “i.e.” They used the approach to the extraction of rhetorical structure presented by Sumita et al. (1993), which groups sentences into tree of relations. For instance, one can distinguish the structure of examples, parallel argument (Firstly, Secondly, Then…), specialization (This is particularly true of…), and so on. The rhetorical structure acquired in this manner can then be compressed by removing entire sentences belonging to be a certain rhetorical category, such as specialization for example (Batang, 2006).

Tagalog-English code-switching (TECS) is widely accepted among bilingual speakers in the Philippines and communities around the world. Despite the prevalence of this phenomenon, there has been limited work focusing on this language pair in the code switching literature. The code-switching literature has been dominated by language pairs that are typologically similar. Further study on typologically dissimilar pairs such as Tagalog and English will be extremely valuable in understanding the mechanisms underlying code-switching (Labitigan, 2013).

Code-switching, one particular phenomenon of bilingual speech, refers to instances of alternating between two languages or varieties of the same language in the same conversation (Myers Scotton, 1983). The linguistic research on code-switching can be generally grouped according to two approaches: structural and sociolinguistic. The structural approach seeks to characterize how code-switching can be represented in the mind. The sociolinguistic approach views code-switching as a socio pragmatic phenomenon, focusing on the social motivations and functions of code-switching (Amuda, 1994). Although both of these main perspectives are invaluable in order to fully understand code-switching, this paper focuses on the grammatical structure of codeswitching (Labitigan, 2013).

Tagalog-English code-switching(TECS), or Taglish, is a variety of bilingual speech. Although it can be considered a prestige language variety, TECS has a increasingly substantial presence in all socioeconomic classes in the Philippines, particularly in urban centers. TECS is also widely spoken by bilingual communities throughout the world (Labitigan, 2013).

It is well known among linguists that nouns are the most code-switched or borrowed forms. In the case of TECS, the nominal domain, which is the subject of this paper, provides many interesting phenomena that require explanation. For the remainder of the paper, I will often be using the term nominal phrase when referring to constituents in the nominal domain, that is, constituents headed by an N. Nominal phrase is a noncommittal term that helps our analysis in two ways. First, nominal phrases of English and of Tagalog seem to behave very differently. Nominal phrases in Tagalog remain a challenging topic of study, and there still remain many competing ideas about their structure and properties. Thus, a general term such as nominal phrase when referring to Tagalog relieves our analysis of some unnecessary complexity. Second, different types of English nominal phrases (i.e. Ns, modified NPs, conjoined NPs, DPs) seem to sometimes pattern together in TECS (for example, see section 3.2.3.1). Thus, an umbrella term helps to capture certain patterns in the data (Labitigan, 2013).

Unlike many other approaches to code-switching, the Matrix Language Frame (MLF) Model (Myers-Scotton, 1993 [1997]) is not simply a collection of descriptive constraints. Rather, it is a multi-layer model with interconnected parts that not only describes linguistic phenomena, but also provides an explanation for why these phenomena occur. At the core of the MLF Model are two key oppositions based on asymmetries in code-switching structures: the Matrix Language (ML) – Embedded Language (EL) opposition and the content-system morpheme opposition (Labitigan, 2013).

The first opposition stems from the fact that the languages involved in codeswitching do not participate equally. A higher level of participation does not refer to a greater number of morphemes or even the presence of certain morphemes, but rather the contribution of more abstract structure. The language that contributes more abstract structure can be referred to as the Matrix Language (ML), while the other language can be referred to as the Embedded Language (EL). The unit of analysis of the MLF Model is the CP (projection of complementizer). A CP is the highest projection of the clause. This unit of analysis for the MLF Model does not only account for the data regarding distributions of the two participating languages, but it also provides an easily identifiable and consistent unit for comparisons across examples and languages. Referring to the CP also allows us to avoid the technical difficulty in defining and distinguishing among other terms such as sentence, clause, and utterance in our analysis. For each CP, there is a grammatical frame specified. This frame, called the ML ,is abstract in nature; it does not itself include any actual morphemes, but rather, “it includes specifications about slots and how they are to be filled, based on directions from lemmas in the mental lexicon” (Myers-Scotton, 2002, p. 67). There is quantitative evidence suggesting that the ML cannot switch within a CP (Finlayson et al., 1998), making the CP the appropriate atomic structure for the study of code-switching. In monolingual speech, the ML frame of each CP is “vacuously transparent” (Jake et al., 2002, p. 72) since the frame is provided by the speaker’s only language. In bilingual speech, this frame may be provided by either one of the two participating languages or, in certain types of contact phenomena, by a combination of the two (Labitigan, 2013).

Unlike many other models, the MLF Model is lexically based. That is, rather than relying solely on principles of monolingual phrase structure to develop accounts for codeswitching, the model underscores abstract procedures in and related to the mental lexicon. Some of these procedures involve phrase structure, but also include other abstract levels. The Abstract Level Model, largely stemming from psycholinguistic models for language production (Levelt, 1993), was developed by Myers-Scotton and Jake (2000) as a supporting model to the MLF Model, but can also stand alone as a description of the levels of abstract lexical structure. The Abstract Level Model designates three levels of abstract lexical structure: lexical-conceptual structure, predicate-argument structure, and morphological realization patterns. A lemma, or an entry in the mental lexicon that maps abstract structure to surface realizations, is represented at all three levels. Thus, the Abstract Level Model serves to trace the path of a linguistic utterance from its beginnings as abstract structure to its manifestation as surface structure. The origin of an utterance comes from an abstract bundle of language independent speaker intentions. These intentions activate an abstract entity known as the Conceptualizer, which refines the message and decides what information is to be communicated linguistically and para-linguistically. The Conceptualizer triggers semantic/pragmatic feature bundles, and the ones that are language-specific are then mapped onto lemmas in the mental lexicon. This mapping forms the first level of abstract lexical structure, lexical-conceptual structure. Once lemmas are active, their morphosyntactic properties (or instructions) can be accessed by the Formulator in order to generate hierarchical morphosyntactic structures. This requires two levels of structure, which involve the language-specific encoding or structural assignment of relations between content morphemes. The first of these two levels, predicate-argument structure, deals with how thematic structure maps onto grammatical relations, and then morphological realization patterns deal with how grammatical relations map onto surface structures (Labitigan, 2013).

One of the main aspects of the so-called “Web 2.0” is increased participation by website users, or a blurring of the distinction between the content provider and the content receiver. One form that this user interaction can take is the sharing of comments on products that users have purchased or services that they have used. Examples abound on websites such as amazon.com, flixster.com, and chapters.indigo.ca. The need for efficient and effective multi-document summarization of these user reviews and other kinds of evaluative text containing opinions and preferences is thus ever-growing (Cheung, 2008).

There are two main approaches to the task of summarization—extraction and abstraction (Hahn and Mani, 2000). Extraction involves concatenating extracts taken from the corpus into a summary, whereas abstraction involves generating novel sentences from information extracted from the corpus. It has been observed that in the context of multi-document summarization of news articles, extraction may be inappropriate because it may produce summaries which are overly verbose or biased towards some sources (Barzilay et al., 1999). However, there has been little work identifying specific factors which might affect the performance of each strategy in summarizing evaluative documents containing opinions and preferences, such as customer reviews or blogs. This chapter aims to address this gap by exploring one dimension along which the effectiveness of the two paradigms could vary; namely, the controversiality of the opinions contained in the corpus (Cheung, 2008).

# CHAPTER 3

**RESEARCH METHODOLOGY**

This chapter includes the steps on how the system will be done and how it will work, also the type of respondents , the type of sampling technique to be used and the instruments that will be used on the study and the data gathering procedure.

## 3.1 Methodology

The researchers will first research and gain a thorough understanding of the domain they are after, which is the online news article summarization. They will procure letters of request for them to be able to conduct surveys and ask for professional assistance from the faculty in the Collage of Arts and Letters in the Polytechnic University of the Philippines in the evaluation of the research’s output. The researchers will also collect a given number of news articles dated January 2014 to September 2014 from Rappler, Philippine daily inquirer and the Philippine star. This will serve as the Population for the study. Meanwhile, the 4th year students and Faculty from the College of arts and letters in PUP will serve as the population for the respondents.

The researchers will then proceed on programming the proposed application. This will be accomplished using Python platform. Other researchers of this study will then search for tools that may help in Natural Language Processing and Understanding, the overall look and feel of the web app, and finally the Generation of the summary itself, the Natural Language Generation. After programming is done, or at least a base implementation of a module is done, it is then tested for quality assurance. This repeats over and over until the researchers are satisfied.

After quality is deemed satisfactory by the researchers, integration, deployment and testing of the application is then done, using the Population of the study which contains a corpus of news articles from January 2014 to September 2014.

Data gathering will be done again, now focused on evaluating the summaries generated by the system. The evaluation consists of testing for the Compression ratio, Understandability, Cohesion, Retention Ratio and Q&A task performance.

Understandability and Cohesion will be evaluated using Coh-Metrix and applying the statistical treatment weighted mean to the results. Compression Ratio is an intrinsic property that will be automatically evaluated by the system. Retention Ratio and Q&A task performance however will need manual evaluation by an expert.

The researchers then tabulate and analyze the data, from which the researchers will derive the conclusion and thus publish the results.

## 3.2 System Architecture

The system takes a News article pasted on the text area or text file as its input. The input will then undergo the preprocessing part where the contents will be put through the sentence split, POS tagging, Language determination, and Phrase-Chunking. The preprocessed data then goes through information item extraction which is a subject-verb-object triple representing a piece in of information in a sentence (Semantic Representation). The resulting representation is then interpreted and fed into the Natural Language generator which tries to generate a sentence or two from the given triples and corresponding phrase tags. The resulting sentences then goes through scoring to pick the most suitable sentence for the summary (Semantic Graph Reduction, Interpretation and Natural Language Generation). **Fig3** shows the architectural representation of the system.



**Figure 3 System Architecture**

## 3.3 Research Paradigm

The researchers will use iterative and incremental development on the study (shown on **Fig 4**). Iterative and incremental development is breaking down software development into a smaller chunks and uses phasing to satisfy software requirements.

The first thing that should be done is to plan what should be the elements needed for the development of the software. Determining the elements, what will be the tools needed for the system work and to finish it will be the next step. Analysis and design is visualizing the model throughout the development of the software cycle. In this phase, the implementation of the structure of news articles will be studied and applied to the coding. After implementing the structure of news articles, the researchers will test the system to know if it is working properly and efficiently. Then the researchers will evaluate if the results from the testing will be acceptable or not. If it is not, the cycle starts again by planning what should be needed to be done to fix the problems encountered until it will satisfy the objectives of the study. After all the software requirements will be met and the software development cycle will be finished, the software will now be ready for deployment.

**Figure 4 Iterative and Incremental Development**

## 3.4 Population of the Study

The Population of the study consists of all online news articles in news websites from Rappler, Inquirer.net, and the Abante.com. The sampling frame that will be used for this study consists of all online news articles from September 2014 to February 2015 news. Since these News Articles are considered recent.

The Population of the Respondents consists of personnel from the Polytechnic University of the Philippines. The sampling frame that will be used for the respondents consists of senior students currently enrolled in the Polytechnic University of the Philippines, and selected faculty member from the College of Arts and Letters.

## 3.5 Types of Sampling Technique

The sampling technique to be used for this study is the **Simple random sampling** technique. Simple random sampling chooses each sampling unit randomly and entirely by chance. Such that each individual has the same probability of being chosen at any stage during the sampling process and each subset of k individuals have the same probability of being chosen for the sample as any other subset of k individuals. (Yates, Moore, & Starnes). This way, the articles chosen for summarization is completely random.

## 3.6 Data Gathering Procedure

### 3.6.1 Preliminaries

The researchers identify what are the problems stated and what will be the possible solutions to solve it. It also involves knowing who will be benefited after finishing the project. After the researchers recognize the factors in solving the problem and what will be the benefits of the system to the user, the data gathering can be made now.

First, the researchers pinpoint the circumstances to solve the problem by identifying what are the needs of the system to work properly and the needs of the user to meet the expected output. Second, finding a suitable and reliable source news articles. Third, identifying the benefits of the system to the user as well as to the community. By finding the benefits, the researchers gathered information on how will it affect to the community and to the user. The researchers also will identify who will take the surveys in order to know significance of the study. Fourth, the researchers identify what tools are needed in order to solve the problem stated.

### 3.6.2 Experimental Method

After identifying the problems to be solved, who will be the user and what benefits can be obtained, data gathering can be made. The researchers will find possible solutions to the problems stated on the related topics. The researchers will also identify the benefits of the system to the user and to the community by identifying the factors affecting each other. With the help of expert(s), the researchers will give experiment papers to compute features of the system. The results of each experiment papers will help the researchers compute measurements of features that will show how accurate and reliable the output will be. Researchers will use Coh-Metrix to measure cohesion and coherence of summary and with the results using Coh-Metrix all gathered researchers can get the average results and get the total cohesion and coherence level of the system. Given the significance of the coherence assumption in the theory, it would be important to dissect and possibly automate coherence and cohesion. (Graesser, McNamara, Louwerse, & Cai, 2004)By giving surveys to a group of people the researchers will classify what are their speculations and what can the researchers do for improvement. In order to solve the problem stated, the researchers will select tools that are needed to the system. The requirement also includes what software and what platform is to be used by the system.

## 3.7 Research Instruments

The researchers will provide questionnaires for population to be used to gather information from Individuals that read news on line or from newspapers, each questionnaires indicates questions to answer and weigh in several questions about the research. The researchers will use the web based system Coh-Metrix to measure the cohesion and cohesiveness of the summarized news article. Gantt chart was also used as a clerical instrument, the Gantt chart states schedule of activities by the researchers. Activities and deadlines like when things need to be done and what should be done today are all placed on several cells showing activity progress per month. Undone activities are marked red and once done it'll be marked green and yellow if it is currently in progress and lastly black if the activity is dropped down or taken off. Gantt chart helped the researchers to be on date on things. Mechanical Instrument used is Python for the programming part. Django will be used for Web designing part where Web functions will be connected using Python. The researchers are also Pythonliterates that make it easier for the researchers to manipulate the edges in Python programming. The database of the research will be accessed using MySQL. Git will be used to sub version files of the system and to let the researchers access file changes and added files easily.

## 3.8 Statistical Treatment

To assess the generated summaries of the system, the following formulas and treatments were used:

1. The **Compression ratio** is the property of a summary that shows how much shorter the summary is than the original, denoted by:

Here, the compression ratio is further subdivided into two parts: length of compression ratio *by the number of sentence s*and *by the number of words.*

**2. Retention ratio** is the property of a summary that shows how much information is retained in the summary from the original, denoted by:

To assess the system’s ability to convey key facts of the source article, the Q&A Task performance evaluation is going to be used, in which experts are asked to read through the source document and marking central passages. Questions that correspond to certain factual statements in the central passages are then made.(Hassel M. , Evaluation of Automatic Text Summarization, 2004) After which the questions are then answered using the system generated summary, judged as *Correct, Partially Correct,* and *missing.*(Mani)

**3. Answer Recall Lenient (ARL)** - ARL is an accuracy metric for the *Q&A task*. ARL measures partially correct and correct answers that will be answered by an expert denoted by:

Where

* **n1** is the number of *Correct* answers
* **n2** the number of *Partially Correct* answers
* **n3** being the number of questions.

**4. Answer Recall Strict (ARS) -** ARS is an accuracy metric for the *Q&A task*. ARS measures and takes only correct answers that will be answered by an expert denoted by:

Where

* **n1** is the number of *Correct* answers
* **n3** being the number of questions.

**5. Answer Recall Average** is accuracy metric for the *Q&A task*, which is the average of the **ARL** and **ARS.** (Mani)

**6.** A **Mean** is the measure of the central tendency of a probability distribution.



Where

* is the summation of frequency
* **N** is the number of respondents
* **N** is the number of respondents

7. **Flesch–Kincaid Grade Level** – Flesch–Kincaid Grade Level is a readability formula that measures *Coherence* of an input article and shows the result in U.S. gradeschool level. The higher the number, the harder it is to read the text. Formula 2 specifies how this score is computed.

Where

* **ASL** is the Average Sentence Length
* **ASW** is the Average number of Syllables per Word

8. Necessary Sample Size – determines the total number of *sample size* that will be use for testing purposes if the population is unknown

Where

* **Margin of Error** - the confidence interval determines how much higher or lower than the population means you are willing to let your sample mean fall.
* **Confidence Level (Z-Score)** - How confident do you want to be that the actual mean falls within your confidence interval
* **Standard of Deviation** **(StdDev)** - How much variance the researchers expect on the responses

For the evaluation of the summaries, the **Sample mean** for each of the abovementioned formulas (*Compression Ratio, Retention Ratio, Answer Recall Strict, Answer Recall Lenient and Answer Recall Average)*.

# 

# CHAPTER 4

**PRESENTATION, ANALYSIS AND INTERPRETATION OF DATA**

The study aims to summarize news articles written in Filipino and English using abstraction methods under Semantic-based summarization. The ratings can be categorized by the Compression ratio, Retention ratio, Coherence, Cohesion and Q&A task performance.

This chapter shows how the researchers interpreted the results of the data gathered from the findings of the experts and the generated summaries. It illustrates the compression of the articles to reduce the length of the content and as well as retaining the thought of the summarized text

**Table 1 Table of Population Size**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Language of Input** | **Margin of Error** | **Confidence Level** | **Standard of Deviation** | **Result** | **Sample Size** |
| **English** | 10% | 90% | .5 | 67.6506 | 68 |
| **Filipino** | 15% | 85% | .5 | 23.04 | 23 |

**Table 4.1** shows the total number of sample size needed to perform implementation. The researchers used the formula of determining necessary sample size to determine the number of news article needed to do implementation. Researchers differs the finding of Filipino and English sample size. The margin of error for English is 10% as the researchers knew how tools used in English are reliable enough to implement a good output, however in Filipino the margin of error is 15% as tools used in Filipino like lexicon, morphology etc. were made by the researchers themselves. The confidence level for both Filipino and English were intervals of error rate that’s why the result in English is 90% and in Filipino the confidence level of 85%. The standard deviation used is .5 as it is the safe one to be used. The resulting number of sample size in English is 68 news articles and in Filipino is 23. It shows that English has more sample size compared to Filipino as the latter has a higher error rate.

**Table 2 Table of Number of Words in English**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **Original** | **Summary** | **Ratio** |
| English1 | 331 | 49 | 0.1480 |
| English2 | 400 | 37 | 0.0925 |
| English3 | 206 | 26 | 0.1262 |
| English4 | 284 | 64 | 0.2254 |
| English5 | 460 | 111 | 0.2413 |
| English6 | 733 | 81 | 0.1105 |
| English7 | 373 | 58 | 0.1555 |
| English8 | 117 | 21 | 0.1795 |
| English9 | 490 | 76 | 0.1551 |
| English11 | 728 | 102 | 0.1401 |
| English12 | 433 | 64 | 0.1478 |
| English13 | 85 | 8 | 0.0941 |
| English14 | 1039 | 171 | 0.1646 |
| English15 | 608 | 97 | 0.1595 |
| English16 | 326 | 32 | 0.0982 |
| English17 | 772 | 120 | 0.1554 |
| English18 | 663 | 100 | 0.1508 |
| English19 | 713 | 124 | 0.1739 |
| English20 | 684 | 104 | 0.1520 |
| English21 | 601 | 102 | 0.1697 |
| English22 | 749 | 145 | 0.1936 |
| English23 | 396 | 55 | 0.1389 |
| English24 | 631 | 76 | 0.1204 |
| English26 | 239 | 32 | 0.1339 |
| English27 | 321 | 49 | 0.1526 |
| English28 | 525 | 98 | 0.1867 |
| English29 | 526 | 67 | 0.1274 |
| English30 | 431 | 57 | 0.1323 |
| English31 | 819 | 147 | 0.1795 |
| English32 | 264 | 39 | 0.1477 |
| English33 | 363 | 80 | 0.2204 |
| English34 | 153 | 25 | 0.1634 |
| English35 | 324 | 38 | 0.1173 |
| English36 | 209 | 34 | 0.1627 |
| English37 | 823 | 153 | 0.1859 |
| English38 | 97 | 31 | 0.3196 |
| English39 | 587 | 89 | 0.1516 |
| English40 | 652 | 87 | 0.1334 |
| English41 | 271 | 24 | 0.0886 |
| English42 | 413 | 46 | 0.1114 |
| English43 | 758 | 115 | 0.1517 |
| English44 | 414 | 63 | 0.1522 |
| English45 | 579 | 97 | 0.1675 |
| English46 | 233 | 27 | 0.1159 |
| English47 | 365 | 48 | 0.1315 |
| English48 | 492 | 88 | 0.1789 |
| English49 | 922 | 135 | 0.1464 |
| English50 | 667 | 90 | 0.1349 |
| English51 | 411 | 59 | 0.1436 |
| English52 | 403 | 64 | 0.1588 |
| English53 | 770 | 118 | 0.1532 |
| English54 | 476 | 76 | 0.1597 |
| English55 | 372 | 67 | 0.1801 |
| English56 | 527 | 90 | 0.1708 |
| English57 | 662 | 95 | 0.1435 |
| English58 | 230 | 18 | 0.0783 |
| English59 | 343 | 49 | 0.1429 |
| English60 | 849 | 149 | 0.1755 |
| English61 | 145 | 10 | 0.0690 |
| English62 | 391 | 61 | 0.1560 |
| English63 | 564 | 86 | 0.1525 |
| English64 | 425 | 74 | 0.1741 |
| English65 | 285 | 53 | 0.1860 |
| English66 | 297 | 72 | 0.2424 |
| English67 | 467 | 63 | 0.1349 |
| English68 | 226 | 32 | 0.1416 |
| English69 | 744 | 85 | 0.1142 |
|  |  |  |  |
| **Average English Word Count** | | | 0.1416 |

**Table 4.2** shows the number of words from the English articles and the generated summaries of the system to get the compression ratio of words. To get the compression ratio of the words, divide the number of words of an English article by the number of words of generated summary. Results of compression per word shows that the system scored a fair score in compressing source input in words. The compression per word scores low as the researchers only takes the Subject-Verb-Object of the source sentence and scores them. The high resulting SVO’s were the only ones that will be listed in the output summarized text. The result may vary if a word on the source input is not listed on the lexicon therefore the word may not be scored for SVO scoring that might result the word to not be included in the output. The resulting average of Word count in English is 0.1416 and shows a low average

**Table 3 Table of Number of Sentences in English**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **Original** | **Summary** | **Ratio** |
| English1 | 17 | 6 | 0.3529 |
| English2 | 15 | 6 | 0.4000 |
| English3 | 10 | 4 | 0.4000 |
| English4 | 21 | 8 | 0.3810 |
| English5 | 26 | 13 | 0.5000 |
| English6 | 29 | 11 | 0.3793 |
| English7 | 16 | 8 | 0.5000 |
| English8 | 7 | 3 | 0.4286 |
| English9 | 26 | 10 | 0.3846 |
| English11 | 31 | 12 | 0.3871 |
| English12 | 23 | 12 | 0.5217 |
| English13 | 3 | 1 | 0.3333 |
| English14 | 53 | 23 | 0.4340 |
| English15 | 25 | 13 | 0.5200 |
| English16 | 17 | 5 | 0.2941 |
| English17 | 35 | 16 | 0.4571 |
| English18 | 29 | 13 | 0.4483 |
| English19 | 43 | 15 | 0.3488 |
| English20 | 28 | 14 | 0.5000 |
| English21 | 29 | 11 | 0.3793 |
| English22 | 33 | 19 | 0.5758 |
| English23 | 14 | 5 | 0.3571 |
| English24 | 27 | 14 | 0.5185 |
| English26 | 12 | 6 | 0.5000 |
| English27 | 9 | 4 | 0.4444 |
| English28 | 44 | 16 | 0.3636 |
| English29 | 28 | 10 | 0.3571 |
| English30 | 14 | 7 | 0.5000 |
| English31 | 42 | 17 | 0.4048 |
| English32 | 14 | 6 | 0.4286 |
| English33 | 18 | 12 | 0.6667 |
| English34 | 8 | 3 | 0.3750 |
| English35 | 13 | 6 | 0.4615 |
| English36 | 11 | 5 | 0.4545 |
| English37 | 42 | 14 | 0.3333 |
| English38 | 6 | 4 | 0.6667 |
| English39 | 39 | 13 | 0.3333 |
| English40 | 29 | 12 | 0.4138 |
| English41 | 8 | 3 | 0.3750 |
| English42 | 17 | 5 | 0.2941 |
| English43 | 40 | 17 | 0.4250 |
| English44 | 15 | 9 | 0.6000 |
| English45 | 25 | 12 | 0.4800 |
| English46 | 14 | 4 | 0.2857 |
| English47 | 12 | 6 | 0.5000 |
| English48 | 30 | 15 | 0.5000 |
| English49 | 38 | 19 | 0.5000 |
| English50 | 124 | 13 | 0.1048 |
| English51 | 16 | 6 | 0.3750 |
| English52 | 16 | 7 | 0.4375 |
| English53 | 52 | 18 | 0.3462 |
| English54 | 26 | 8 | 0.3077 |
| English55 | 19 | 9 | 0.4737 |
| English56 | 28 | 11 | 0.3929 |
| English57 | 41 | 14 | 0.3415 |
| English58 | 8 | 4 | 0.5000 |
| English59 | 13 | 7 | 0.5385 |
| English60 | 47 | 21 | 0.4468 |
| English61 | 7 | 2 | 0.2857 |
| English62 | 15 | 6 | 0.4000 |
| English63 | 23 | 12 | 0.5217 |
| English64 | 27 | 9 | 0.3333 |
| English65 | 12 | 8 | 0.6667 |
| English66 | 11 | 7 | 0.6364 |
| English67 | 18 | 8 | 0.4444 |
| English68 | 14 | 5 | 0.3571 |
| English69 | 54 | 11 | 0.2037 |
|  |  |  |  |
| **Average English Sentence Count** | | | 0.4151 |

Table 4.3 shows the number of sentences from the English articles and the generated summaries of the system to get the compression ratio of sentences. To get the compression ratio of the sentences, divide the number of sentences of an English article by the number of sentences of generated summary. The ratio resulted a fair score of 0.4151.

**Table 4 Table of number of Words in Filipino**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **Original** | **Summary** | **Ratio** |
| Filipino1 | 159 | 47 | 0.2956 |
| Filipino2 | 188 | 17 | 0.0904 |
| Filipino5 | 234 | 3 | 0.0128 |
| Filipino6 | 554 | 35 | 0.0632 |
| Filipino7 | 161 | 8 | 0.0497 |
| Filipino8 | 265 | 41 | 0.1547 |
| Filipino9 | 251 | 12 | 0.0478 |
| Filipino11 | 203 | 9 | 0.0443 |
| Filipino12 | 154 | 19 | 0.1234 |
| Filipino14 | 159 | 8 | 0.0503 |
| Filipino15 | 196 | 29 | 0.1480 |
| Filipino16 | 159 | 47 | 0.2956 |
| Filipino22 | 157 | 30 | 0.1911 |
| Filipino24 | 167 | 20 | 0.1198 |
| Filipino26 | 144 | 11 | 0.0764 |
| Filipino27 | 112 | 7 | 0.0625 |
| Filipino28 | 175 | 8 | 0.0457 |
| Filipino29 | 258 | 2 | 0.0078 |
| Filipino30 | 234 | 38 | 0.1624 |
| Filipino31 | 141 | 7 | 0.0496 |
| Filipino32 | 109 | 20 | 0.1835 |
| Filipino34 | 230 | 10 | 0.0435 |
| Filipino35 | 322 | 39 | 0.1211 |
|  |  |  |  |
| **Average Filipino Word Count** | | | 0.0813 |

The table shows the number of words from the Filipino articles and the generated summaries of the system to get the compression ratio of words. To get the compression ratio of the words, divide the number of words of Filipino article by the number of words of generated summary.

**Table 5 Table of number of Sentences in Filipino**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **Original** | **Summary** | **Ratio** |
| Filipino1 | 10 | 4 | 0.4000 |
| Filipino2 | 9 | 2 | 0.2222 |
| Filipino5 | 7 | 1 | 0.1429 |
| Filipino6 | 23 | 8 | 0.3478 |
| Filipino7 | 8 | 1 | 0.1250 |
| Filipino8 | 11 | 5 | 0.4545 |
| Filipino9 | 11 | 3 | 0.2727 |
| Filipino11 | 7 | 1 | 0.1429 |
| Filipino12 | 11 | 1 | 0.0909 |
| Filipino14 | 6 | 1 | 0.1667 |
| Filipino15 | 8 | 3 | 0.3750 |
| Filipino16 | 11 | 4 | 0.3636 |
| Filipino22 | 5 | 3 | 0.6000 |
| Filipino24 | 9 | 3 | 0.3333 |
| Filipino26 | 6 | 1 | 0.1667 |
| Filipino27 | 4 | 1 | 0.2500 |
| Filipino28 | 12 | 11 | 0.9167 |
| Filipino29 | 12 | 1 | 0.0833 |
| Filipino30 | 11 | 4 | 0.3636 |
| Filipino31 | 9 | 2 | 0.2222 |
| Filipino32 | 5 | 3 | 0.6000 |
| Filipino34 | 6 | 2 | 0.3333 |
| Filipino35 | 14 | 6 | 0.4286 |
|  |  |  |  |
| **Average Filipino Sentence Count** | | | 0.2467 |

The table shows the number of sentences from the Filipino articles and the generated summaries of the system to get the compression ratio of words. To get the compression ratio of the words, divide the number of words of Filipino article by the number of words of generated summary.

**Table 6 Table of Retention in English**

|  |  |
| --- | --- |
| **Filename** | **Ratio of Retained Information** |
| English2 | 0.6666 |
| English3 | 0.5 |
| English4 | 0.6666 |
| English5 | 1 |
| English6 | 1 |
| English7 | 0.4 |
| English8 | 0.5 |
| English9 | 0.2857 |
| English11 | 0.6 |
| English12 | 0.25 |
| English13 | 0.3333 |
| English14 | 0.5 |
| English15 | 0.2 |
| English16 | 0.4 |
| English17 | 0.4 |
| English18 | 1 |
| English19 | 0.8 |
| English20 | 0.5 |
| English21 | 0.8 |
| English22 | 1 |
| English23 | 0.4 |
| English24 | 0 |
| English26 | 0.3333 |
| English27 | 0.6 |
| English28 | 0.2 |
| English29 | 0.5 |
| English30 | 0.8333 |
| English31 | 0.8571 |
| English32 | 0.5 |
| English33 | 0.5714 |
| English34 | 0.6 |
| English35 | 0.6 |
| English36 | 0 |
| English37 | 0.3333 |
| English38 | 0.75 |
| English39 | 0.8 |
| English40 | 0.3 |
| English41 | 0.25 |
| English42 | 0.3333 |
| English43 | 0.375 |
| English44 | 0.5 |
| English45 | 0.2222 |
| English46 | 0.5 |
| English47 | 0.2 |
| English48 | 0.2857 |
| English49 | 0.25 |
| English50 | 0.625 |
| English51 | 0.3333 |
| English52 | 0.3333 |
| English53 | 0.3333 |
| English54 | 0.5 |
| English55 | 0.5 |
| English56 | 0.6 |
| English57 | 0.25 |
| English58 | 0.1666 |
| English59 | 0.6 |
| English60 | 0.4 |
| English61 | 0.3333 |
| English62 | 0.1666 |
| English63 | 0.5714 |
| English64 | 0.4285 |
| English65 | 0.75 |
| English66 | 0.6666 |
| English67 | 0.6 |
| English68 | 0.4 |
| English69 | 0.75 |
|  |  |
| **Mean** | 0.4588 |

**Table 7 Table of Retention in Filipino**

|  |  |
| --- | --- |
| **Filename** | **Ratio of Retained Information** |
| Filipino1 | 0.4 |
| Filipino2 | 0.25 |
| Filipino5 | 0.3333 |
| Filipino6 | 0.1428 |
| Filipino7 | 0.2 |
| Filipino8 | 0.5 |
| Filipino9 | 0 |
| Filipino11 | 0.3333 |
| Filipino12 | 0.25 |
| Filipino14 | 0.3333 |
| Filipino15 | 0.25 |
| Filipino16 | 0.4 |
| Filipino22 | 0.3333 |
| Filipino24 | 0.6666 |
| Filipino26 | 0.25 |
| Filipino27 | 0 |
| Filipino28 | 0.2 |
| Filipino29 | 0 |
| Filipino30 | 0.2 |
| Filipino31 | 0.3333 |
| Filipino32 | 0 |
| Filipino34 | 0.25 |
| Filipino35 | 0.2 |
|  |  |
| **Mean** | 0.1941 |

**Table 8 Table of Coherence and Cohesion in English**

|  |  |  |
| --- | --- | --- |
| **Filename** | **Cohesion** | **Coherence** |
| English2 | 11% | 9.16 |
| English3 | 23% | 13.9 |
| English4 | 12% | 7.03 |
| English5 | 3% | 9.78 |
| English6 | 4% | 10.62 |
| English7 | 10% | 10.5 |
| English8 | 89% | 6.4 |
| English9 | 8% | 10.8 |
| English11 | 22% | 10.7 |
| English12 | 9% | 9.7 |
| English13 | 55% | 9.1 |
| English14 | 8% | 11.6 |
| English15 | 5% | 10.2 |
| English16 | 5% | 5 |
| English17 | 10% | 8.8 |
| English18 | 17% | 9.6 |
| English19 | 10% | 12.2 |
| English20 | 2% | 8.5 |
| English21 | 14% | 12.2 |
| English22 | 2% | 12.5 |
| English23 | 2% | 11.3 |
| English24 | 4% | 6.9 |
| English26 | 11% | 10 |
| English27 | 2% | 12.1 |
| English28 | 3% | 4.7 |
| English29 | 4% | 10.4 |
| English30 | 4% | 10.9 |
| English31 | 2% | 9.8 |
| English32 | 2% | 9.6 |
| English33 | 5% | 12.1 |
| English34 | 15% | 9.3 |
| English35 | 35% | 12.3 |
| English36 | 3% | 8.9 |
| English37 | 2% | 10.62 |
| English38 | 25% | 8.02 |
| English39 | 4% | 9.79 |
| English40 | 2% | 8.27 |
| English41 | 23% | 3.43 |
| English42 | 2% | 8.03 |
| English43 | 15% | 7.51 |
| English44 | 7% | 8.67 |
| English45 | 8% | 8.67 |
| English46 | 2% | 11.27 |
| English47 | 2% | 6.59 |
| English48 | 68% | 5.31 |
| English49 | 2% | 11.083 |
| English50 | 2% | 7.65 |
| English51 | 2% | 12.07 |
| English52 | 10% | 6.74 |
| English53 | 11% | 4.35 |
| English54 | 35% | 10.43 |
| English55 | 37% | 13.69 |
| English56 | 2% | 12.17 |
| English57 | 13% | 8.92 |
| English58 | 2% | 9.37 |
| English59 | 54% | 6.13 |
| English60 | 6% | 12.76 |
| English61 | 2% | 5.14 |
| English62 | 6% | 14.66 |
| English63 | 3% | 7.41 |
| English64 | 5% | 10.36 |
| English65 | 17% | 10.13 |
| English66 | 15% | 13.53 |
| English67 | 2% | 7.29 |
| English68 | 31% | 6.18 |
| English69 | 9% | 5.39 |
|  |  |  |
| **Total** | 12% | 9.091514706 |

**Table 9 Table of Question and Answer average in English**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **ARL** | **ARS** | **ARA** |
| English2 | 0.3333 | 0.1666 | 0.2499 |
| English3 | 0.3333 | 0 | 0.1666 |
| English4 | 0.3 | 0 | 0.15 |
| English5 | 0.4166 | 0.3333 | 0.3749 |
| English6 | 0.7 | 0.4 | 0.55 |
| English7 | 0.3 | 0 | 0.15 |
| English8 | 0.375 | 0.25 | 0.3125 |
| English9 | 0.3571 | 0.1428 | 0.2499 |
| English11 | 0.3 | 0 | 0.15 |
| English12 | 0.0833 | 0 | 0.0416 |
| English13 | 0.0833 | 0 | 0.0416 |
| English14 | 0.35 | 0.1 | 0.45 |
| English15 | 0.1 | 0 | 0.05 |
| English16 | 0.0833 | 0 | 0.0416 |
| English17 | 0.25 | 0.1666 | 0.2083 |
| English18 | 0.0357 | 0.2857 | 0.3214 |
| English19 | 0.4285 | 0.1428 | 0.2856 |
| English20 | 0.25 | 0 | 0.125 |
| English21 | 0.4285 | 0.2857 | 0.3571 |
| English22 | 0.6666 | 0.5 | 0.5833 |
| English23 | 0.3 | 0 | 0.15 |
| English24 | 0.1666 | 0 | 0.0833 |
| English26 | 0.2857 | 0.1428 | 0.2142 |
| English27 | 0.5714 | 0.4285 | 0.4999 |
| English28 | 0.3333 | 0.1666 | 0.4999 |
| English29 | 0.2857 | 0 | 0.1428 |
| English30 | 0.375 | 0.125 | 0.25 |
| English31 | 0.375 | 0.25 | 0.3125 |
| English32 | 0.1666 | 0 | 0.0833 |
| English33 | 0.4444 | 0.3333 | 0.3888 |
| English34 | 0.25 | 0.1666 | 0.2083 |
| English35 | 0.5 | 0.3333 | 0.8333 |
| English36 | 0.25 | 0 | 0.125 |
| English37 | 0.3333 | 0.3333 | 0.3333 |
| English38 | 0.0833 | 0 | 0.041 |
| English39 | 0.1666 | 0.1666 | 0.1666 |
| English40 | 0.4166 | 0.3333 | 0.3749 |
| English41 | 0.5 | 0.5 | 0.5 |
| English42 | 0 | 0 | 0 |
| English43 | 0.4 | 0 | 0.2 |
| English44 | 0.375 | 0.25 | 0.3125 |
| English45 | 0.3333 | 0.1666 | 0.2499 |
| English46 | 0.375 | 0.25 | 0.3125 |
| English47 | 0.5 | 0.25 | 0.375 |
| English48 | 0.375 | 0.25 | 0.3125 |
| English49 | 0.25 | 0 | 0.125 |
| English50 | 0.25 | 0.25 | 0.25 |
| English51 | 0.25 | 0 | 0.125 |
| English52 | 0.3 | 0.2 | 0.25 |
| English53 | 0.4166 | 0.1666 | 0.2916 |
| English54 | 0.4 | 0.2 | 0.3 |
| English55 | 0.4166 | 0.3333 | 0.2399 |
| English56 | 0.7 | 0.6 | 0.65 |
| English57 | 0.1 | 0 | 0.05 |
| English58 | 0.1428 | 0 | 0.0714 |
| English59 | 0.375 | 0.25 | 0.3125 |
| English60 | 0.375 | 0.25 | 0.3125 |
| English61 | 0 | 0 | 0 |
| English62 | 0 | 0 | 0 |
| English63 | 0.5 | 0.25 | 0.375 |
| English64 | 0.625 | 0.5 | 0.5626 |
| English65 | 0.3 | 0.2 | 0.25 |
| English66 | 0.5 | 0 | 0.25 |
| English67 | 0.3 | 0.2 | 0.25 |
| English68 | 0.1666 | 0 | 0.0833 |
| English69 | 0.375 | 0 | 0.1875 |
|  |  |  |  |
| **Total** | 0.31 | 0.15617 | 0.24655 |

**Table 10 Table of Question and Answer average in Filipino**

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename** | **ARL** | **ARS** | **ARA** |
| Filipino1 | 0.125 | 0 | 0.0625 |
| Filipino2 | 0 | 0 | 0 |
| Filipino5 | 0 | 0 | 0 |
| Filipino6 | 0.2 | 0.2 | 0.2 |
| Filipino7 | 0 | 0 | 0 |
| Filipino8 | 0.25 | 0 | 0.125 |
| Filipino9 | 0 | 0 | 0 |
| Filipino11 | 0.25 | 0.25 | 0.25 |
| Filipino12 | 0.5 | 0.3333 | 0.4166 |
| Filipino14 | 0.3333 | 0.3333 | 0.3333 |
| Filipino15 | 0 | 0 | 0 |
| Filipino16 | 0.3333 | 0.3333 | 0.3333 |
| Filipino22 | 0.3333 | 0 | 0.1666 |
| Filipino24 | 0.5 | 0.5 | 0.5 |
| Filipino26 | 0 | 0 | 0 |
| Filipino27 | 0 | 0 | 0 |
| Filipino28 | 0.3333 | 0 | 0.1666 |
| Filipino29 | 0 | 0 | 0 |
| Filipino30 | 0.375 | 0.25 | 0.3125 |
| Filipino31 | 0.25 | 0.25 | 0.25 |
| Filipino32 | 0.25 | 0 | 0.125 |
| Filipino34 | 0 | 0 | 0 |
| Filipino35 | 0 | 0 | 0 |
|  |  |  |  |
| Total | 0.13444 | 0.081663333 | 0.108047 |

# CHAPTER 5

**SUMMARY OF FINDINGS, CONCLUSION AND RECOMMENDATIONS**

This chapter includes the results of all the findings made on the study, Conclusions made while doing the study based on data analyzed on the previous chapter. Some limitations have been found and recommendations for future researchers that might help improve the system.

## 5.1 Summary

The purpose of this research was to probe the essence of using graph based abstractive summarization, especially with the use of Chunking for extracting Information Items instead of Dependency parsers, in both English and Filipino languages. Previous works shows the use of extraction summarization and resulting in fair accuracy, but showing that extraction only gets the phrase of the desired input. The study shows ineffectiveness in summarizing Filipino sentences for there are limited Filipino tools that can be used and that the researchers made some on their own (eg. Lexicon & Morphology). An experiment paper was used and was given to an expert to measure the retention of the system and also experts formulated questions and answer task to measure the recall of the system. Cohesion and coherence were measured using Coh-Metrix and the compression ratio was measured by the researchers.

The objective of the study was to measure the ratings of the system using generated summaries. The results were as follows:

a. Compression Ratio – 0.2805

b. Retention Ratio – 0.4771

c. Understandability/Coherence – 9.0915

d. Cohesion - 0.12

e. Q&A Task performance - 0.2465

The results shows that graph based summarization in Filipino are still unreliable and in English that there are dependencies when it comes to contents and structure of each sentence’s words and phrases. The lack of a complete dictionary in Filipino also contributed on the result for if a word was not found on the lexicon the information item/s containing the word would not be included in the candidate sentences for the summary. The system resulted a low compression ratio as the system does not compress the input article much and there times where the system will output the same Subject-Verb-Object of a sentence that will result in duplication. The retention level scores an average result for the system takes the Subject-Verb-Object of the input that will become a candidate for the resulting output; there are times where the system generates non sense sentences or phrases. This happens because the system’s method of obtaining the Subject-Verb-Object triple of each sentence uses Phrase chunking and is reliant on training data and does not support anaphora resolution, thus resulting in triples that make no sense at all. Coherence and Cohesion displayed a low result as the results were generated on the web system Coh-Metrix. By measuring syllables present on sentences Coh-Metrix can compute the Flesch-Kincaid Grade Level of the input, the higher the resulting digit is the harder the sentences is to be read. The Question and Answer task resulted fair as researchers asked an assist from an expert in language in both Filipino and English to conduct the process. The expert reads the input and generated questions that will be answered by the output. Correct answers scores 1 and partially correct scores 0.5 while wrong answers were 0 and with the results researchers computed the ARS and ARL and computed the average using ARA.

## 5.2 Conclusions

According to the experimentation made by the language expert, and the implementation results gathered and computed, the system little of the objectives are met. This result on the lack of tools in Filipino, the poor implementation made, the poor application of the SVO triple extraction and scoring and the poor chunking and part of speech tagging made by the tools used on the system. The poor output can also be affected by the input as the system was made for news format of texts only, and as news articles are always on the passive voice.

## 5.3 Recommendations

After concluding the result of the system, the researchers made the following recommendations might be helpful for researchers to improve the system:

1. Using a complete lexicon of Filipino words to avoid undetected words on the system.
2. More training data both for the English/Filipino POS and Chunk taggers to avoid missing Noun Phrases that should be detected
3. Use of Anaphora resolution to better determine if a verb has an object or if some word is referring to another
4. A port of the language detector in python, as the current used detector requires java

# Bibliography

*Article (Publishing)*. (n.d.). Retrieved from Wikipedia: http://en.wikipedia.org/wiki/Article\_(publishing)

*Automatic Summarization*. (n.d.). Retrieved from Wikipedia: http://en.wikipedia.org/wiki/Automatic\_summarization

Batang, E. (2006). *NewsByes: A Tagalog News Text Summarization by Abstraction.* Philippines.

Cheung, J. (2008). *Comparing Abstractive and Extractive Summarization of Evaluative Text: Controversiality and Content Selection.*

Dalianis, & Hassel. (2000, 2004).

Dalianis, & Hassel. (2001).

Dalianis, H., & Hassel, M. *Generation of Reference summary.*

Davis, G. Sampling. In *Davis/Gallardo's Straight Talk About Communication Research Methods 1st edtion* (p. 164). Kendall Hunt Publishing Corporation.

Delianis, & Hassel. (2005). *Generation of Reference Summaries.*

Delianis, & Hassel. *How Short is Good? An evaluation of automatic summarization.*

Donatelli, J. (2013). *Pay-By-The-Word Journalism is Ridiculous.*

Etienne Genest, P., & Lapalme, G. (n.d.). Text Generation for Abstractive Summarization. 1.

Foltz, P. W. (1996). Latent semantic analysis for text-based research. 198.

Gong, Y., & Liu, X. *Generic Text Summarization Using Relevance and Latent Semantic Analysis.* C & C Research Laboratories USA.

Graesser, A. C., McNamara, D. S., Louwerse, M. M., & Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. 1.

Hahn. (2001). *The Challenges of Automatic Summarization.*

Hassel, M. (2004). *Evaluation of Automatic Text Summarization.*

Hassel, M. *Exploitation of Named Entities in Automatic Text Summarization for Swedish.*

Hassel, M. (2000). *Pronominal Resolution in Automatic Text Summarization.*

Khan, S. (2014). *A Review on Abstractive Summarization Methods.*

Labitigan, R. (2013). *Tagalog-English code-switching: issues in the nominal domain.* Philippines.

Mani. *The TIPSTER SUMMAC Text Summarization Evaluation.*

Pachantouris, G. (2004-2005). *GreekSum.*

Paxton, T. *Daily News.*

Philippines, U. o. (2002). *Online News Summarizer http://www.engg.upd.edu.ph/~naval/cs198/cs198/2002/ons/index.html.*

Sidner, C. (1984). *Focusing in the Comprehension of Definite Anaphora. Berwich& Brady eds., Computational Models of Discourse.* London.

*sra.com.*

Stones, T. R. (1967). *Yesterday's Papers.*

Yates, D., Moore, D. S., & Starnes, D. S. *The Practice of Statistics, 3rd Ed. Freeman.*

# Appendix A

## Graphical User Interface

3

2

1



4

6

5

1. **File tab** for file uploading page where users can upload txt files from their computer that contains news articles.
2. **Text tab** this page is the part where users can paste news articles that will be summarized on a Text Area provided.
3. **Text Area** is where the directory or the text will be pasted or input.
4. **Choose file button** for File tab that will trigger the file directory.
5. **Upload Button** for upload tab that will trigger summarization once clicked.

The File Tab has a Choose file button that will find news files from the directory of the user the will serve as an input



File tab that has a chosen file for an input. Once chosen the user can press the Upload button to Summarize the input file and the output summarized news article will be displayed on the Text Area.



Text Tab displays text area where the user may paste a news article that will serve as an input. By clicking the SumMe button the system will summarize the pasted text and the output will be displayed on the text area.



Input news article will be pasted on the text area below. The output will be the summarize version of the pasted input news article. The summarization may take a while, but once finished the user will see the result posted on the text area.



The output on the Text Tab will be displayed on the Text Area. Once summarized a Download button will appear where the user may download the result and save it as a text file for future purposes.



# Appendix B

## Implementation Report

**INTRODUCTION**

SumMe is a Abstractive based summarization that summarized news article in English and Filipino, it is a web based summarization sysem that takes text files or pasted news articles as input on the user interface. SumMe process and display the summarized news article on the GUI and the user will be given an option to save the result summary.

**PROBLEM STATEMENT**

The Researchers needs to measure the performance and functionality of the system and to do this, the researchers will take the compression ratio as a measurement for how much the system can compress the output. Compression Ratio can be measured by the researchers themselves. However, Retention Ratio is the problem, the researchers needs to conduct an experimentation process and needs an advice from an expert to measure how much the system can recall and retain information from the source input. Another is conducting a Question Game to measure the Answer Recall Average of the system. With all the measured information gathered, the researchers can measure the functionality of the system.

**RESPONDENTS/SUBJECTS**

The respondent needed for the experimentation method is an expert in both Filipino and English languages. The number of subjects or sample data was resulted from estimating error rates, z-scoring and confidence level since the number of population size was not determined.

**TIME FRAME**

The schedule of the implementation was two days before the submission of the final paper for the thesis. Before implementation, we talked to the professionals and experts if they were available for testing and evaluating our system's performance in summarizing a news article. We also search and gathered both English and Filipino news articles from the web of different websites. for implementation The duration of testing was 2 days with more than 12 hours each day. The professional that we got tested and evaluated our system for two days.

**IMPLEMENTATION PROCEDURES**

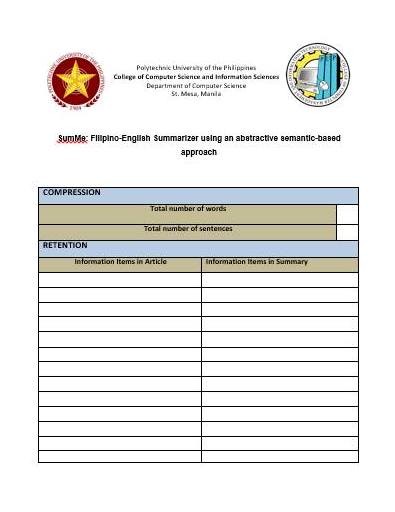
The researchers gathered the news articles from websites like Rappler and Daily Inquirer and with those, the researchers determined the number of sample size needed for implementation. Each sample was tested and the number of words and sentences were listed to determine the compression rate of the system. The researchers asked an advice from an expert to answer experiment papers provided by the researchers to review summarized articles and list retrieved information on each for every experiment paper. A Q/A also took place where the expert read inputs and layout questions that should be answered by the summary. With those information the researchers computed the total ARA and Retention Ratio of each test data. Cohesion and Coherence were measured by a web system named Coh-Metrix. The resulting cohesion and coherence were listed with the other gathered information to compute the functionality of the system.

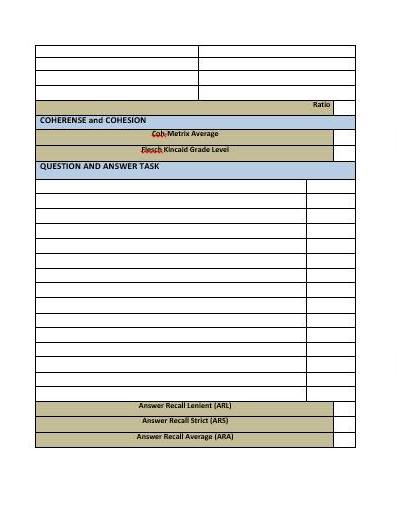
**ISSUES AND CONCERNS**

The system cannot process unicode characters, so if a unicode character is present on a news article file the system will result in an error that took the researchers more time to implement the system.

# Appendix C

## Experiment Paper





# Appendix D

## Letter for Expert



# Appendix E

## Photos of Implementation with Expert





# Appendix F

## Computation of the Results of the Study

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Filename** | **CW** | **CS** | **Retention** | **Cohesion** | **Coherence** | **ARL** | **ARS** | **ARA** |
| English2 | 0.0925 | 0.4000 | 0.6666 | 11% | 9.16 | 0.3333 | 0.1666 | 0.2499 |
| English3 | 0.1262 | 0.4000 | 0.5 | 23% | 13.9 | 0.3333 | 0 | 0.1666 |
| English4 | 0.2254 | 0.3810 | 0.6666 | 12% | 7.03 | 0.3 | 0 | 0.15 |
| English5 | 0.2413 | 0.5000 | 1 | 3% | 9.78 | 0.4166 | 0.3333 | 0.3749 |
| English6 | 0.1105 | 0.3793 | 1.24 | 4% | 10.62 | 0.7 | 0.4 | 0.55 |
| English7 | 0.1555 | 0.5000 | 0.4 | 10% | 10.5 | 0.3 | 0 | 0.15 |
| English8 | 0.1795 | 0.4286 | 0.5 | 89% | 6.4 | 0.375 | 0.25 | 0.3125 |
| English9 | 0.1551 | 0.3846 | 0.2857 | 8% | 10.8 | 0.3571 | 0.1428 | 0.2499 |
| English11 | 0.1401 | 0.3871 | 0.6 | 22% | 10.7 | 0.3 | 0 | 0.15 |
| English12 | 0.1478 | 0.5217 | 0.25 | 9% | 9.7 | 0.0833 | 0 | 0.0416 |
| English13 | 0.0941 | 0.3333 | 0.3333 | 55% | 9.1 | 0.0833 | 0 | 0.0416 |
| English14 | 0.1646 | 0.4340 | 0.5 | 8% | 11.6 | 0.35 | 0.1 | 0.45 |
| English15 | 0.1595 | 0.5200 | 0.2 | 5% | 10.2 | 0.1 | 0 | 0.05 |
| English16 | 0.0982 | 0.2941 | 0.4 | 5% | 5 | 0.0833 | 0 | 0.0416 |
| English17 | 0.1554 | 0.4571 | 0.4 | 10% | 8.8 | 0.25 | 0.1666 | 0.2083 |
| English18 | 0.1508 | 0.4483 | 1 | 17% | 9.6 | 0.0357 | 0.2857 | 0.3214 |
| English19 | 0.1738 | 0.3488 | 0.8 | 10% | 12.2 | 0.4285 | 0.1428 | 0.2856 |
| English20 | 0.152 | 0.5000 | 0.5 | 2% | 8.5 | 0.25 | 0 | 0.125 |
| English21 | 0.1697 | 0.3793 | 0.8 | 14% | 12.2 | 0.4285 | 0.2857 | 0.3571 |
| English22 | 0.1936 | 0.5758 | 1 | 2% | 12.5 | 0.6666 | 0.5 | 0.5833 |
| English23 | 0.1389 | 0.3571 | 0.4 | 2% | 11.3 | 0.3 | 0 | 0.15 |
| English24 | 0.1204 | 0.5185 | 0 | 4% | 6.9 | 0.1666 | 0 | 0.0833 |
| English26 | 0.1339 | 0.5000 | 0.3333 | 11% | 10 | 0.2857 | 0.1428 | 0.2142 |
| English27 | 0.1526 | 0.4444 | 0.6 | 2% | 12.1 | 0.5714 | 0.4285 | 0.4999 |
| English28 | 0.1867 | 0.3636 | 0.2 | 3% | 4.7 | 0.3333 | 0.1666 | 0.4999 |
| English29 | 0.1274 | 0.3571 | 0.5 | 4% | 10.4 | 0.2857 | 0 | 0.1428 |
| English30 | 0.1323 | 0.5000 | 0.8333 | 4% | 10.9 | 0.375 | 0.125 | 0.25 |
| English31 | 0.1795 | 0.4048 | 0.8571 | 2% | 9.8 | 0.375 | 0.25 | 0.3125 |
| English32 | 0.1477 | 0.4286 | 0.5 | 2% | 9.6 | 0.1666 | 0 | 0.0833 |
| English33 | 0.2204 | 0.6667 | 0.5714 | 5% | 12.1 | 0.4444 | 0.3333 | 0.3888 |
| English34 | 0.1634 | 0.3750 | 0.6 | 15% | 9.3 | 0.25 | 0.1666 | 0.2083 |
| English35 | 0.1173 | 0.4615 | 0.6 | 35% | 12.3 | 0.5 | 0.3333 | 0.8333 |
| English36 | 0.1627 | 0.4545 | 0 | 3% | 8.9 | 0.25 | 0 | 0.125 |
| English37 | 0.1859 | 0.3333 | 0.3333 | 2% | 10.62 | 0.3333 | 0.3333 | 0.3333 |
| English38 | 0.1389 | 0.6667 | 0.75 | 25% | 8.02 | 0.0833 | 0 | 0.041 |
| English39 | 0.1516 | 0.3333 | 0.8 | 4% | 9.79 | 0.1666 | 0.1666 | 0.1666 |
| English40 | 0.1334 | 0.4183 | 0.3 | 2% | 8.27 | 0.4166 | 0.3333 | 0.3749 |
| English41 | 0.0886 | 0.3750 | 0.25 | 23% | 3.43 | 0.5 | 0.5 | 0.5 |
| English42 | 0.1114 | 0.2941 | 0.3333 | 2% | 8.03 | 0 | 0 | 0 |
| English43 | 0.1517 | 0.4250 | 0.375 | 15% | 7.51 | 0.4 | 0 | 0.2 |
| English44 | 0.1522 | 0.6000 | 0.5 | 7% | 8.67 | 0.375 | 0.25 | 0.3125 |
| English45 | 0.1675 | 0.4800 | 0.2222 | 8% | 8.67 | 0.3333 | 0.1666 | 0.2499 |
| English46 | 0.1159 | 0.2857 | 0.5 | 2% | 11.27 | 0.375 | 0.25 | 0.3125 |
| English47 | 0.1315 | 0.5000 | 0.2 | 2% | 6.59 | 0.5 | 0.25 | 0.375 |
| English48 | 0.1789 | 0.5000 | 0.2857 | 68% | 5.31 | 0.375 | 0.25 | 0.3125 |
| English49 | 0.1464 | 0.5000 | 0.25 | 2% | 11.083 | 0.25 | 0 | 0.125 |
| English50 | 0.1349 | 0.1048 | 0.625 | 2% | 7.65 | 0.25 | 0.25 | 0.25 |
| English51 | 0.1436 | 0.3750 | 0.3333 | 2% | 12.07 | 0.25 | 0 | 0.125 |
| English52 | 0.1588 | 0.4375 | 0.3333 | 10% | 6.74 | 0.3 | 0.2 | 0.25 |
| English53 | 0.1532 | 0.3462 | 0.3333 | 11% | 4.35 | 0.4166 | 0.1666 | 0.2916 |
| English54 | 0.1597 | 0.3077 | 0.5 | 35% | 10.43 | 0.4 | 0.2 | 0.3 |
| English55 | 0.1801 | 0.4737 | 0.5 | 37% | 13.69 | 0.4166 | 0.3333 | 0.2399 |
| English56 | 0.1708 | 0.3929 | 0.6 | 2% | 12.17 | 0.7 | 0.6 | 0.65 |
| English57 | 0.1435 | 0.3415 | 0.25 | 13% | 8.92 | 0.1 | 0 | 0.05 |
| English58 | 0.0783 | 0.5000 | 0.1666 | 2% | 9.37 | 0.1428 | 0 | 0.0714 |
| English59 | 0.1429 | 0.5385 | 0.6 | 54% | 6.13 | 0.375 | 0.25 | 0.3125 |
| English60 | 0.1755 | 0.4468 | 0.4 | 6% | 12.76 | 0.375 | 0.25 | 0.3125 |
| English61 | 0.069 | 0.2857 | 0.3333 | 2% | 5.14 | 0 | 0 | 0 |
| English62 | 0.156 | 0.4000 | 0.1666 | 6% | 14.66 | 0 | 0 | 0 |
| English63 | 0.1525 | 0.5217 | 0.5714 | 3% | 7.41 | 0.5 | 0.25 | 0.375 |
| English64 | 0.1741 | 0.3333 | 0.4285 | 5% | 10.36 | 0.625 | 0.5 | 0.5626 |
| English65 | 0.186 | 0.6667 | 0.75 | 17% | 10.13 | 0.3 | 0.2 | 0.25 |
| English66 | 0.2424 | 0.6364 | 0.6666 | 15% | 13.53 | 0.5 | 0 | 0.25 |
| English67 | 0.1349 | 0.4444 | 0.6 | 2% | 7.29 | 0.3 | 0.2 | 0.25 |
| English68 | 0.1416 | 0.3571 | 0.4 | 31% | 6.18 | 0.1666 | 0 | 0.0833 |
| English69 | 0.1142 | 0.2037 | 0.75 | 9% | 5.39 | 0.375 | 0 | 0.1875 |
|  |  |  |  |  |  |  |  |  |
| **Total** | 0.1461 | 0.4151 | 0.47712794 | 12% | 9.091514706 | 0.31 | 0.15617 | 0.24655 |

# Appendix G

## Vitae



Justin Allen Capalad was born on May 31, 1993 at Quezon City, Metro Manila. He graduated on Felix Memorial National High School of Cainta, Rizal and is currently taking up BS Computer Science at Polytechnic University of the Philippines, Manila.



Renz Gio A. Evangelista was born on December 29, 1994 at Marikina City, Metro Manila. He graduated on Sta. Elena High School of Marikina City and is currently taking up BS Computer Science at Polytechnic University of the Philippines, Manila.