CORPUS-DRIVEN MALAY LANGUAGE TWEET NORMALIZATION

MOHAMMAD ARSHI SALOOT

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2018

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Mohammad Arshi Saloot (I.C/Passport No: [redacted])

Matric No: WHA110062

Name of Degree: PhD in Computer Science

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"): CORPUS-DRIVEN MALAY LANGUAGE TWEET NORMALIZATION

Field of Study: Computer Science/Artificial Intelligence

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

CORPUS-DRIVEN MALAY LANGUAGE TWEET NORMALIZATION

ABSTRACT

The expeditious spread of blogs, microblogs, and social network services has led to accelerate the usage of casual written language, known as user generated content (UGC). The UGC diverges from standard writing conventions because of the usage of coding strategies, such as phonetic transcriptions (*are* → *r*), digit phonemes (*me too* → *me2*), misspellings (*misappropriate* → *missapropriate*), vowel drops (*double* → *dble*), and missing or incorrect punctuation marks (*In that situation, I'd possibly come.* → *In that situation Id possibly come*). These modifications are due to three primary elements: 1) limited message length (e.g. 140 characters per Tweet); 2) miniature keyboards; and 3) extensive usage of UGC in unofficial and informal communications. However, the existence of many out-of-vocabulary (OOV) words, also known as unknown words, substantially disturbs standard natural language processing (NLP) systems. Therefore, research in NLP has increasingly focused on the text normalization task, where the OOV words will convert into their context-appropriate standard words. Currently, while diverse normalization approaches exist in the English language, the problem is neglected in other languages, such as Malay language. In this work, the Malay language is chosen because of its considerable usage on Twitter, where, it is the fourth leading language used in Twitter. Thus, a rule-based approach to normalize the Malay language Twitter messages is proposed based on corpus-driven analysis. To do so, a corpus-driven analysis depends on frequencies in specifying word-frequency lists, concordancing, clusters, and keywords. To design the normalization system, three analyzing tasks on the Malay language Twitter corpus and standard Malay corpus were performed: 1) frequency of unknown words; 2) abbreviation patterns; and 3) letter repetition. A Malay language Twitter corpus known as Malay Chat-style Corpus (MCC) is constructed. The MCC, which encompasses 1 million twitter messages, consists of

14,484,384 word instances, 646,807 unique vocabularies, and metadata, such as used Twitter client application, posting time, and type of Twitter message (simple Tweet, Retweet, Reply). To build the MCC, which represents the Malay language Twitter lingo, corpus-compiling criteria were considered which are: sampling, representativeness, machine readability, balance, and size of data. A portion of the MCC is manually annotated to be used in the development and testing stages of the normalization system. The architecture of the Malay normalization system contains seven primary modules: (1) enhanced tokenization; (2) In-Vocabulary (IV) detection; (3) colloquial dictionary lookup; (4) repeated letter elimination; (5) abbreviation normalizer; (6) English word translation; and (7) de-tokenization. The normalization modules are formulated based on the result of MCC analysis and implemented via rule-based state machines. An evaluation is performed in term of BLEU score to measure the accuracy of the system. The result is encouraging whereby 0.91 BLEU score is achieved against 0.46 BLEU baseline score. To compare the accuracy of the system with other probabilistic approaches with an identical Malay dataset, statistical machine translation (SMT) normalization system is chosen to be implemented, trained, and evaluated. The experimental results prove that higher accuracy is achieved by the proposed architecture, which is designed based on the results of our corpus-driven analysis.

Keywords: Corpus, Normalization, Malay, UGC

NORMALISASI TWEET BAHASA MELAYU DIDORONG KORPUS

ABSTRAK

Penggunaan blog, mikroblog, dan perkhidmatan rangkaian sosial yang semakin meluas telah membawa kepada perkembangan penggunaan bahasa tulisan yang bersifat kasual, yang dikenali sebagai kandungan yang dijana pengguna (UGC). UGC berbeza dari piawai penulisan konvensyen kerana banyak menggunakan strategi pengekodan termasuk perubahan dalam transkripsi fonetik (*are → r*), fonem angka (*me too → me2*), salah ejaan (*misappropriate → missapropriate*), pengguguran vokal (*double → dble*), dan tanda bacaan yang hilang atau salah (*In that situation, I'd possibly come. → In that situation Id possibly come*). Pengubahsuaian ini adalah disebabkan oleh tiga elemen utama, iaitu: 1) mesej yang terhad (contoh 140 aksara setiap Tweet); 2) papan kekunci yang kecil; dan 3) penggunaan UGC yang meluas dalam komunikasi tidak rasmi dan tidak formal. Walau bagaimanapun, kewujudan banyak perkataan di luar kosa kata *"Out-of-Vocabulary"* (OOV), mengganggu sistem piawai pemprosesan bahasa tabii (NLP). Dalam normalisasi teks, sistem akan menggantikan perkataan OOV dengan perkataan piawai yang sesuai dengan konteks. Pada masa ini, pendekatan normalisasi dalam bahasa Inggeris sudah menjadi perkara lazim, namun masalah tersebut masih lagi tidak diberi perhatian dalam bahasa-bahasa lain termasuklah bahasa Melayu. Bahasa Melayu menjadi tumpuan kerana penggunaannya yang luas dalam Twitter, dan merupakan bahasa keempat terbanyak yang digunakan di Twitter. Dalam kajian ini, pendekatan berasaskan peraturan untuk dicadangkan menormalisasikan mesej bahasa Melayu di laman Twitter berdasarkan analisis korpus. Analisis berasaskan korpus bergantung kepada frekuensi dalam menentukan frekuensi senarai perkataan, konkordans, kelompok dan kata kunci. Untuk mereka bentuk sistem normalisasi, tiga analisis ke atas korpus bahasa Melayu di laman Twitter dan korpus piawai bahasa Melayu telah dijalankan iaitu: 1) kekerapan perkataan yang tidak diketahui; 2) corak

singkatan perkataan; dan 3) pengulangan huruf. Sehubungan dengan itu, satu korpus Twitter bahasa Melayu yang dikenali sebagai "*Malay Chat-style Corpus"* (MCC). MCC, yang merangkumi 1 juta mesej Twitter, yang terdiri daripada 14.484.384 contoh perkataan, 646.807 terma dan metadata, seperti aplikasi pelanggan Twitter digunakan, mencatat masa, dan jenis Twitter mesej. Untuk membina MCC, yang mewakili Twitter bahasa Melayu, korpus beberapa kriteria korpus diambil kira: kerepresentatifan "*representativeness*", persampelan, keseimbangan, kebolehbacaan usia, dan saiz data. Sebahagian daripada MCC adalah dianotasi secara manual yang akan digunakan dalam peringkat pembangunan dan sistem pengujian normalisasi. Sistem normalisasi bahasa Melayu yang dibina mengandungi tujuh modul utama, iaitu: (1) peningkatan token "*enhanced tokenization*"; (2) Dalam kosa kata "*In-Vocabulary detection*"; (3) carian kamus spesifik "colloquial dictionary lookup*"*; (4) penghilangan huruf yang diulang "*repeated letter elimination"*; (5) penyelarasan singkatan "*abbreviation normalizer"*; (6) terjemahan perkataan bahasa Inggeris "*English word translation*"; and (7) pengurangan token "*de-tokenization*". Untuk mengukur ketepatan sistem, ujian telah dijalankan. Hasilnya menunjukkan skor sebanyak 0.91 dalam BLEU terhadap garis dasar BLEU, dan hanya menunjukkan skor sebanyak 0.46. Untuk membandingkan ketepatan sistem dengan pendekatan statistik yang lain, sistem normalisasi perterjemahan mesin statistik (seperti SMT) dilaksanakan, dilatih, dan dinilai dengan set data yang sama. Keputusan eksperimen menunjukkan bahawa sistem normalisasi berasaskan peraturan, yang dibina berdasarkan ciri-ciri Tweet bahasa Melayu mencapai ketepatan yang menggunakan analisis berdasarkan korpus.

Kata Kunci: Korpus, Normalisasi, Melayu, UGC

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Dr. Norisma Idris for the continuous support of my PhD study, for her patience, motivation, and immense knowledge.

Besides my advisor, I would like to thank all UM academic and non-academic staffs specially Dr. Rohana Mahmud, Dr. Salinah Jaafar, Prof. Datin Dr. Sameem Binti Abdul Kareem, Mr. Mazrul and Mrs. Norazarina Bohari, for their encouragements and insightful comments.

Thanks to you, reader. If you are reading this line after the others, you at least read one page of my thesis. Thank You.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI: Artificial Intelligence

ASR: Automatic speech recognition

BRC: Borneo Research Council

CRF: Conditional Random Field

CC: Colon Character

DBP: Dewan Bahasa dan Pustaka

DC: Dot Character

EOL: End-of-Line

EM: Exclamation Mark

GUI: Graphic User Interface

HMM : Hidden Markov Model

IDE: Integrated Development Environment

IDF: Inverse Document Frequency

IDLE: Integrated Development and Learning Environment

IE: Information Extraction

IR: Information Retrieval

IV: In Vocabulary

IW: In-vocabulary Word

LCS: Longest Common Subsequence

LM: Language Model

LP: Left Parenthesis

LQM: Left Quotation Mark

MCC: Malay Chat-style Corpus

MT: Machine Translation

MWU: Multi-Word Units

NER: Named Entity Recognition

NIST: National Institute of Standards and Technology

NLP: Natural Language Processing

NPMI : Normalized Pointwise Mutual Information

NSW: Non Standard Word

NT: Normalized Token

OOV: Out of Vocabulary

PC: Personal Computer

PN: Proper Noun

POS: Part of Speech

RE: Regular Expressions

RP: Right Parenthesis

RQM: Right Quotation Mark

SMS: Short Message Service

SMT: Statistical Machine Translation

SNS: Social Network Services

TTS: Text-to-Speech

TF: Term Frequency

TF–IDF: Term Frequency – Inverse Document Frequency

UGC: User Generated Content

WER: Word Error Rate

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Introduction

Computer-Mediated-Communication (CMD) is a type of communication or conversation between two or more individuals that takes place over computer networks. The awareness of the Internet facilitates modern telecommunication and services such as instant messaging, emails, and chat rooms to both private and public sectors. Moreover, it is generally known that the combination of both Social Network Services (SNS) and Weblogs are a revolutionary generation of CMD. An example of this combination is Twitter website. Twitter is one of the most popular micro-blogging website with over 500 million registered users as of 2012 (Dugan, 2012).

Currently, Twitter has become a rich source for various projects such as Artificial Intelligence (AI) and Information Retrieval (IR) projects. Twitter Corpus can be used for predicting election using sentiment analysis and opinion mining techniques (Tumasjan et al,. 2010), which is one of the most popular research in recent year. In addition, Twitter can also be used in academic or informal student collaborative e-learning (Vivian, Barnes, Geer, & Wood, 2014).

There are Machine Translation (MT) projects that convert Twitter messages to other languages such as automatically translating English Twitter messages into Arabic language (Jehl, Hieber, & Riezler, 2012). However, one of the main obstacles to get high accuracy is the high usage of the colloquial language of Twitters' users. The presence of ungrammatical text, Out-of-Vocabulary (OOV) words, slang phrases, incorrect and missing punctuations, and short formed words have become an obstacle to perform Information Retrieval (IE) and Natural Language Processing (NLP) tasks for Twitter. Furthermore, mushrooming IE and NLP projects which involve Twitter

corpus motivate researchers to tackle the colloquial language, converting colloquial text to its standard form of language.

Text normalization deals with Non-Standard Words (NSW) such as youth generated words, Uniform Resource Locators (URLs) and E-mails, and short formed words. The objectives of the conventional text normalization include word tokenization, sentence tokenization, digit and number detection, and proper noun detection. Text normalization solutions attack the problem of the conventional spell checkers in the untidy text domain. For example, Kiss and Strunk, (2006) formulated the following simple rule for detecting the end of sentence in the conventional manner as:

Any sequence of alphabets or digits that end with a period and followed by a space and again followed by a sequence of letters or digits, which begin with majuscule letter indicate the end of the sentence.

Whereas in Tweets, to detect the end of a sentence, a user will not begin the next sentence with a majuscule letter or a user will not leave a space between two sentences.

In addition, recent researches on text normalization attempt to deal with Internet slangs, youth slangs, shortened form of words, and many other features in chat-style texts. However, most of the researches on Twitter considered only English language. Though Twitter messages written in Malay and Indonesian languages per day are about 6%, which is approximately 3 million messages per day as 2013 (Hsu & Lin, 2013), presently, there are only few researches Twitter messages conducted in Malay language. Therefore, this research focuses on the twittering in Malay language, which is the fourth leading language used in Twitter.

1.2 Research Motivation

Converting Twitter lingo to standard Malay language has several benefits that includes:

- Machine Translation (MT) tools can easily convert the normalized messages to other languages. Without normalizing, Twitter messages MT tools find many OOV words which do not exist in their dictionary or in their parallel corpus.

- Recently, Twitter stance has one of the most significant resources for performing sentiment analysis projects. For instance, many companies used Twitter to understand customers' feelings about their products in the stock market (Choudhury et al., 2007). It is also used for estimating public sentiment for political elections. However, achieving high accuracy in sentiment analysis is very difficult because some Twitter messages contain a lot of OOV words and some words do not have sentence period.

- Also, there are a lot of research interests and projects on performing Information Extraction (IE) based on Twitter. A good example of IE project is Popescu's research on detecting and understanding events and event's audience (Popescu, Pennacchiotti, & Paranjpe, 2011). Though ill-formed words and intentionally misspelled words are the major challenges in IE projects.

1.3 Problem Statement

User Generated Content (UGC) such as Twitter messages contain a lot of OOV words. UGC content is different from the conventional ways of writings. According to experts (Bieswanger, 2007 and Thurlow & Brown, 2003), the difference is as a result of the usage of various coding strategies such as digit phonemes (e.g. *you too* → *you2*),

phonetic transcriptions (e.g. *you* → *u*), vowel drops (e.g. *dinner* → *dnnr*), misspellings (e.g. *convenience* → *convineince*), and missing or incorrect punctuation marks (e.g. *If I were you, I'd probably go.* →*If I were you Id probably go*).

Most of the NLP applications, such as POS taggers and named entity recognizers are developed to work with standard language. The existence of OOV words stands as a challenge when applying NLP applications to UGC texts. Presently, there are several researches on English normalization, in both supervised and unsupervised learnings using AI methods (Beaufort, Roekhaut, Cougnon, & Fairon, 2010; Cook & Stevenson, 2009 and Pennell & Liu, 2011a). Most spoken languages in the world like French, Spanish, and Chinese have motivated a lot of researchers to design their own normalization systems (Beaufort et al., 2010; Montero & Lorenzo, 2011 and Wang, Kan, Andrade, Onishi, & Ishikawa, 2013). There are few researches on text normalization in Malay language (Basri, Alfred, & On, 2012; Samsudin, Puteh, Hamdan, & Nazri, 2012). A research on the effects of text normalization in social media by Baldwin and Li, (2015), suggested how normalization task should be viewed and it should be highly dependent on the targeted application. Therefore, studies on the normalization for the Malay language Tweets are necessary.

1.4 Aims and Objectives of Research

The aim of this research is to design a normalization architecture for Malay language Twitter messages which can convert ungrammatical colloquial-style text with unknown words (OOV words) to the standard Malay text. The text message normalization aims to replace the non-standard tokens that carry significant meaning in the context-appropriate standard words. The design of the normalization system is based on analysis of both the standard Malay language and Malay language Twitter messages.

The objectives of this research work are to:

1) Compile a corpus that represents the colloquial Malay language in Twitter.

2) Analyze colloquial language and standard language corpora. Malay unknown words and their features, including their frequencies and patterns of abbreviations should be identified. The pattern of repeated letters in the standard Malay language should also be identified.

3) Design a Malay language Tweet normalization architecture which can convert the Malay language Tweet lingo into its standard Malay language. An algorithm for each module in the normalization architecture will be formulated based on the results of objective 2.

4) Evaluate the performance of the Malay language Tweet normalization architecture with our benchmark.

1.5 Research Questions

This research work answers the research questions that corresponds to the objectives identified in the section 1.4.

Objective 1. To compile a corpus that represents the colloquial Malay language in Twitter

- How to develop a corpus for the Malay language Twitter messages?
- What are the criteria used to represent the corpus?

Objective 2. To analyze colloquial language and standard language corpora.

- What are unknown words?
- What is the frequency of the unknown words in Malay language Tweets?

- What are the patterns of abbreviations of unknown words in Malay?
- What are the patterns of repeated letters in the standard Malay language?

Objective 3. To design a Malay language Tweet normalization architecture which can convert the Malay language Tweet lingo into its standard Malay language.

- How to design a Malay language Tweet normalization architecture?
- How many modules are there in the architecture?
- How to develop the algorithms of the normalization modules?
- What is the technique used to develop the algorithms?

Objective 4. To evaluate the performance of the Malay language Tweet normalization architecture with our benchmark.

- How is the performance of the architecture?
- How is the performance of the normalization system as compared to the state-of-the-art systems?

1.6 Research Methodology

The research method is divided into five phases as shown in Figure 1.1:

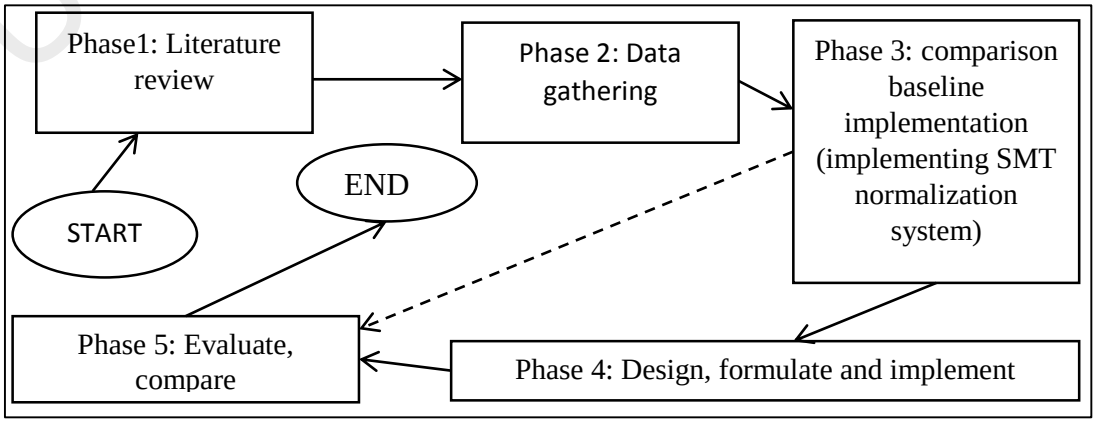

Figure 1.1: Research Methodology

- Phase 1: Understanding problem: In this phase, a comprehensive literature review that is associated directly or indirectly with the problem domain and deep investigation of the previous solutions were conducted.

- Phase 2: Collecting Data: In this phase, electronic textual data were collected in two forms, which are;

    - Compiling a representative Malay chat-style text corpus.

    - Compiling Malay Twitter lingo dictionary.

- Phase 3: Constructing comparison baseline: In this phase, different proposed solutions to normalize Twitter messages are studied. Although, some of those existing solutions depend on the specific language, while some needed more resources such as online abbreviation expanding web services, which is not available in Malay language. However, understanding the strengths and weaknesses existing solutions will serve as a useful tool for developing Malay language by extracting Malay language lingo features. In addition, a Statistical Machine Translation (SMT) normalization system (Aw, Zhang, Xiao, & Su, 2006) is implemented for Malay language Tweets to compare the results with our proposed normalizer in this phase.

- Phase 4: Designing Phase: In this phase, a normalization architecture is implemented. Set of rules for each type of Malay unknown words are formulated. Also, in this phase, heuristic algorithms for our problem domain is formulated.

- Phase 5: Evaluating Phase: This evaluates the performance and results of the proposed architecture is provided and compared our solution with the benchmark. The activities in this phase include testing the functionality of the implemented prototype, and trading off the performance and accuracy of the architecture.

1.7 Thesis Organization

This thesis is organized as follows.

- Chapter 2 gives a basic introduction into the corpus. The chapter also clarifies the terminology used in corpus-based and corpus-driven studies and provides a description of corpus compiling criteria such as data collection, corpus size, and representativeness.

- Chapter 3 describes the previous studies on noisy text normalization. The chapter specifically discussed the existing approaches of the Malay language Tweet normalization.

- Chapter 4 presents an analysis on the Malay language Tweets and also the standard Malay text to determine their features. The chapter also presents several corpora compiling criteria for Malay language Tweet corpus. The main contribution of this chapter is to provide a Malay chat style corpus.

- Chapter 5 presents the proposed normalization architecture. It discussed how rule-based algorithm can be utilized in noisy text normalization architecture. It also demonstrates how the architecture can converts Malay noisy text to standard Malay text.

- Chapter 6 presents the evaluation results of the proposed normalization architecture. The evaluation results are experimenting using two approaches. The first experimental approach is to show the accuracy of the implemented rule-based Malay normalization prototype which is proposed in this research. The second experimental approach is to evaluate the accuracy of the SMT normalization approach (i.e. comparison baseline) on Malay data to compare it with our rule-based architecture.

- Finally, Chapter 7 presents the main conclusion and the main contributions of this research work. It also addresses a few issues that must be taken into consideration for future research.

In summary, the overall structure of the thesis is illustrated diagrammatically in Figure 1.2.

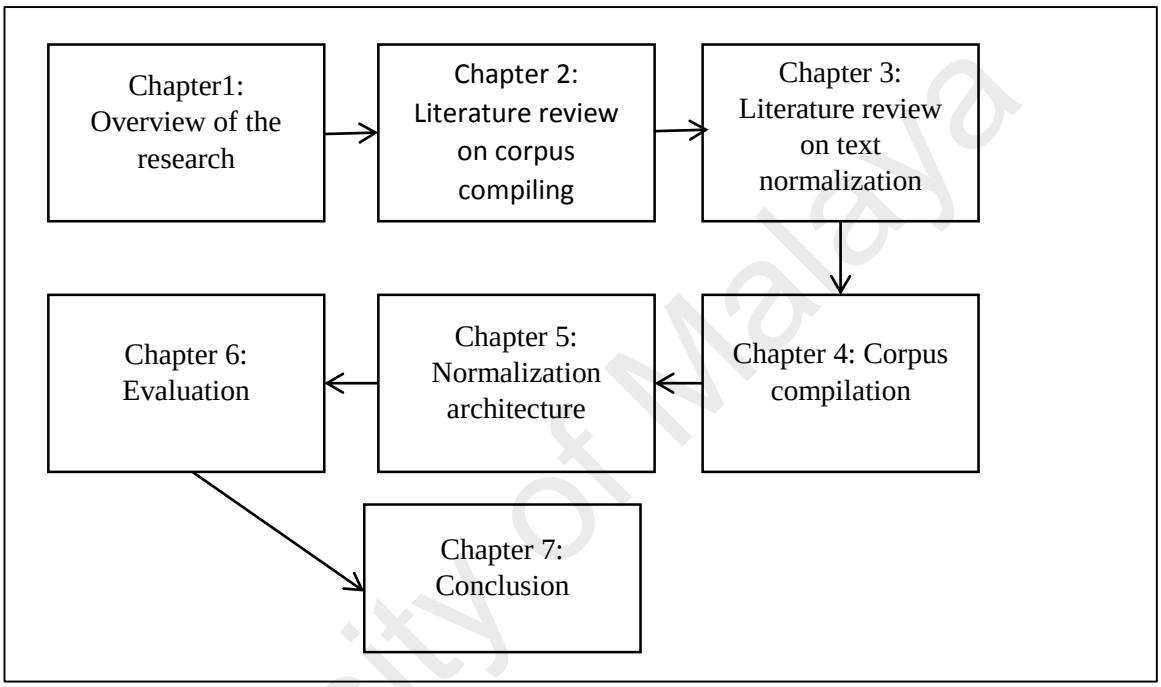

Figure 1.2: Thesis Organization

CHAPTER 2: CORPUS-DRIVEN APPROACH

2.1 Introduction

This chapter is divided into three main sections. The first section describes corpus-driven approach and its differences with corpus-based. Corpus-driven studies try to understand and analyze a language while corpus-based studies try to justify predefined assumptions, theories or ideas about a language (Biber, 2012). However, both approaches need a reliable corpus in order to proceed for better analysis. The second section illustrates criteria of corpus compiling which include corpus size, representativeness, sampling, variety, chronology, transcript errors, and annotation. The third section elaborates our analysis method.

2.2 Corpus-driven Studies

Tognini-Bonelli (2001) presented a dichotomy relation between corpus-based and corpus-driven techniques. The dichotomy relation between the two techniques became an attractive topic of study between corpus linguistics (e.g. McEnery, Xiao, & Tono, 2006, Section A1.7). Figure 2.1 presented the relationship between both techniques.

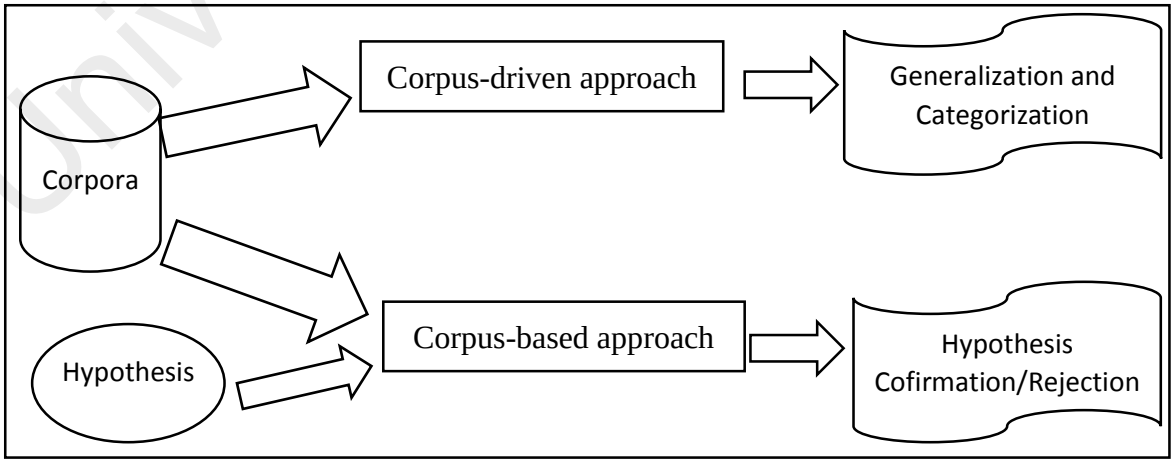

Figure 2.1: Corpus-Driven and Corpus-Based Approaches

Figure 2.1 shows the approaches are separated from each other. However, researches such as, Corpus-informed by (Carter & McCarthy, 2006), corpus-assisted by (Partington, 2006; Stubbs, 1986) and corpus-infused by (Bang, 2009), all provides detailed discussion on both corpus-based and corpus-driven techniques. At the end of this chapter, the similarities of this research with corpus-based or corpus-driven studies would be discussed.

2.2.1 Corpus-based Approach

The major purpose of corpora analysis is linked with the connectivity of both corpus-driven and corpus-based approaches, i.e., both approaches should be associated with their corresponding proponents (Tagg, 2009). Corpora is been utilized by corpus-based research mainly as resource center for the purpose of providing pre-corpus descriptions of language or to illustrate present theories, like that of Tognini-Bonelli, (2001). Although, the techniques used by existing theories differ. However, they all use the corpora tagged features such as, Part-of-Speech (POS), root of words, grammatical feature, frequency analysis, and feature co-occurrence.

The corpus-based analysis can be referred as a set of techniques that it is readily utilized in different applications than corpus-driven. For instance, it is used as systemic functionalists by Matthiessen (2006), and also used by different researchers (Leech, et al., 2009; Mair, et al., 2002) for observing changes in diachronic language. In addition, some language variation studies also claimed to be corpus-based (Sampson, 2003). Corpus-based study is a valuable tool that enables researchers for results validation and theoretical claims. Though, it also claimed that corpus-based approach does not encourage the new language description perception (Tagg, 2009).

2.2.2 Corpus-driven Approach

The corpus-driven analysis cannot be adjusted in any form to suit the predefined hypothesis when the corpus evidence is already provided, This is because corpus-driven analysis is fully consistent and have a direct reflection (Tagg, 2009). In corpus-driven studies, patterns are developed from the data itself (Tognini-Bonelli, 2001), and these data have been approached with clear notions by corpus-driven linguists. Although, it is usually observed that the traditional grammatical types like POS and clause components are not constantly suitable for a natural-occurring data, which create room for new descriptions of language (Tagg, 2009). In reality, this simply indicates that there is a need to consider the following in corpus-driven approach: frequencies dependence, multi-word recognition (phrases), keywords, concordance, and descriptive power of lexicon-grammatical patterning (Sinclair, 1991, 2004). In addition, there is no specific technique for data extraction in corpus-driven approach, but data should be grounded in certain principles regarding corpus compiling (Tagg, 2009).

The corpus-driven approach has guided researchers in developing the following: pattern grammars by Hunston and Francis (2000), lexis and phrases using language-learning syllabuses by Paran (1993) and the COBUILD by Boguraev (1990). It is also applicable in both scientific and academic disciplines (Groom, 2009).

2.2.2.1 Corpus Analysis

A key question in corpus-driven studies is how to quantify what a text is about by considering the words that build up the text. One measure of how essential a word is term frequency (TF). TF measures the frequency of word appearances in a text. There are high frequency words in a text that are not important. For example, in Malay language, these are stop words such as "*yang*", "*adalah*", and "*daripada*". Usually a list of stop words will be compiled and be removed from the text before analysis.

However, it is possible that some of the high frequency words are more important in some context than others. Removing all stop words is not a mature approach for tuning TF (Aizawa, 2003; W. Zhang, Yoshida, & Tang, 2008).

The more sophisticated approach is to consider term's inverse document frequency (IDF). In the IDF, the weight of high frequency words is declined and the weight of uncommon words in a collection of documents is escalated. IDF can be integrated with TF to achieve term's TF-IDF: the recurrence of a word that is fine-tuned for how bare it is appeared. While the initial intention of TF-IDF was to calculate significance of a word in a single text or in a corpus, it has proved useful in text mining, search engines, and other areas of natural language processing (Aizawa, 2003; W. Zhang et al., 2008). TF-IDF is calculated in Example 2.1 using below formulas:

$$TF - IDF(w, t, T) = TF(w, t).IDF(w, T)$$

$$TF(w, t) = (Number\ of\ times\ word\ w\ appears\ in\ a\ text\ t)/$$
$$(Total\ number\ of\ words\ in\ the\ text\ t)$$

$$IDF(w, T) = \log_e(Total\ number\ of\ text\ T/Number\ of\ text\ \ t\ with\ word\ w\ in\ it)$$

Example 2.1:

A text containing 100 words with three appearances of word *'Abang'*,

10 million texts, including 1000 texts that encompass the word *'Abang'*,

TF = (3 / 100) = 0.03,

IDF = log(10,000,000 / 1,000) = 4,

TF-IDF = 0.03 * 4 = 0.12.

A string pattern recognition is required to apply TF or TF-IDF to the corpus. A regular expression (RE) is an expression utilized to identify a set of strings required. A pattern

in RE will be tested to find matches in a target string. The pattern is formed out of an arrangement of atoms. An atom is a distinct part of the RE pattern that looks for matches in a target text Despite all the formal definition of RE in language theories, it is widely available in the most of programming languages such as Python.

2.3 Corpus Compiling Criteria

Ideally, to have a meaningful textual representation, there are some specific requirements and established criteria that all selected/gathered texts must meet. Sinclair (2004) presented corpus representative dimension that needed to be attained in order to have meaningful texts representation, among are; all texts language should be collected in electronic form, external criteria should be used in selecting texts as a resource of data to represent language or language.

2.3.1 Corpus Size

Corpora comprising of less than one million words are quite small. Corpus size can be small but it must be suitably big enough for its purposes. In addition, corpus generally depend on the following two factors, according to McEnery et al. (2006): 1) Practical considerations and 2) Research focus. Hunston and Francis (2000) stated that the size of specialized corpora is smaller than the general reference corpora, because in language, the general corpora always occupy more space than the specialized corpora. Currently, both specialized corpora and large reference corpora are gaining greater recognition and they are used for specific language analysis (Ghadessy, Henry, & Roseberry, 2001).

A specialized corpus should not be compared to the British National Corpus (BNC) or Cambridge International Corpus (CIC), because both BNC and CIC cover many kinds of varieties. However, corpus size is equally dependent on one of the following two features (Scott, 2001), which are; frequency of the actual studied features, and the

purpose in studying them. For example, in language textbook, there is a need to use large corpora, because compilers are concerned about the new vocabulary learners do encounter frequently. But smaller corpora are needed for obscure or creative hapaxes (Scott, 2001). Furthermore, the growing dimension alongside with the variation in language perceptions makes the corpus size not to be absolute, but the size can be calculated by the number of texts.

The limited accessibility of data makes the concept of stability to be more important than corpus size. The conventional operational approach for defining corpus stability is built on the concept of closure or saturation (McEnery & Wilson, 2001). Corpora are known to be lexically saturated when they are divided into segments and the number of new lexical items does not have effects or change the additional new segments (McEnery et al., 2006).

According to McEnery et al. (2006), saturation deals with how representative corpora are of a lexicon and also the rate of lexical growth. Dahlmann (2007) extended the previous study and uses linguistic features such as; Multi-Word Units (MWUs) to measure corpus stability. In addition, n-grams and Wmatrix are the two techniques that are used for assessing the effects of corpus stability. It is assumed that there are little changes between the first ten MWUs when the datasets are large. However, the maximum corpus size is known when first ten MWUs are in a stable state (Dahlmann, 2007). Dahlmann (2007) used around 25,000 words for the segments when analyzing their MWUs. But they concluded that 200,000 words are large for the analysis. The essence of their research is to show that each corpus are not only accessed its composition but also for its particular purpose.

2.3.2 Representativeness

In general, a corpus should represent a predefined target population. Therefore, corpus representativeness depends on knowing and defining the entire population size that the corpus is proposed to represent and the methods that are used for selecting the sample.

The definition of the target population are categorized into two (Biber, 1993);

(1) The population boundaries: It defined the texts that are included and excluded from the population.

(2) The population hierarchical organization: It defined and grouped included texts in the population.

Unfortunately, when designing text corpora, the above definition is not sufficient enough, because population samples are collected without prior definition of the target population. This makes it difficult to evaluate the efficiency or representativeness of a particular corpus, due to lack of a well-defined conception of what the sample is intended to represent (Biber, 1993).

Furthermore, corpus representativeness depends on the level at which the sample population includes linguistic distributions, i.e. the level at which different linguistic features are differently distributed (within texts, across texts, across text types), and a representative corpus must have the capability of analyzing these distributions (Biber, 1993).

2.3.3 Sampling

Conventionally, the size of the corpora is considered by researchers as the most significant element in the representativeness of corpora (Biber, 1993). Although literatures on sampling theories have also proven that sampling has greater significance in achieving representativeness (Biber, 1993). Sampling processes take place in three

phases, which are; 1) phase one defined target population, 2) phase two specified sample frame, and 3) phase three involved data gathering data based on the sampling approach selected. Defining "Target population" enhance the evaluation process of corpus representativeness making it easier to know the desired population. But the population is represented using sampling frame (McEnery et al., 2006).

There are two major sampling approaches, which are; the probability and non-probability approach. If all elements in a sample frame have a chance of being selected in a sample set, one of suitable probability schemes would be followed such as simple random sampling, systematic sampling, stratified random sampling, or cluster sampling (Biber, 1993). In the non-probability sampling, only a few elements in the sample frame are considered. Choudhury et al. (2010) in their study on the impact of data sampling strategy on Twitter corpus information diffusion, proves that pure context-based sampling techniques such as location-based performs better than random sampling. Purposive sampling also known as judgmental, selective or subjective sampling is a type of non-probability sampling, which selection decisions are formed based on the population of interest (Wattam, 2015). Therefore, to perform a precise selection of samples, purposive sampling scheme has been chosen in this study. For example, in the sample frame, there are users who post their Tweets in English language, thus they should be considered as out-of-coverage.

It is necessary to adopt the sample text that public read (reception) or the text that public write (production) (Atkins, Clear, & Ostler, 1992). The reception language is related to addressing a little part of the entire language. Moreover, it is very difficult to collect production text, because production text is associated with personal emails and other information. Population is representative of a chat-style text. This simply means that chat-style corpus consists of only production text.

2.3.4 Variety and Chronology

It is evident that every author has his or her own style, that is, there are considerable differences in the use of lexis, grammar and discourse features between authors. Therefore, selecting more users influences the coverage of language styles varieties. Also, modern corpus-based analysis only focuses on language that is composed of certain period.

Synchronic corpora record language data collected for one specific point in time. Diachronic corpora collect language for vast periods of time and they deal with the way language develops over time. The dynamic nature of language has made researchers to investigate and carry out research on the changes in language such as identifying if there are any changes in previous language features when viewing the current language features. However, the use of synchronic corpus is mostly the suitable approach for specialized corpora (Renouf, 2002, Section II). So, it is left for corpus compiler to choose whether to develop a synchronic corpus or diachronic based on to its potential users.

2.3.5 Corpus Structure

The internal structure of a final corpus is known through corpus structure specification, although, some criteria needed to be established for data fragmentation. Most importantly, data must be fragmented into different sections/clauses at some particular points using characters such as the period (.), colon (:), or semicolon (;). Thus, the clause boundary must be established when developing a corpus. In addition, in the case of designing diachronic corpora, a long-term data collection strategy should be planned (Ahmad & Mathkour, 2009).

It should be noted that a sub-corpora structure is an important criteria to be considered when defining the initial development phase of the corpus. An appropriate sub-corpora

structure supports and enhance corpus readability, reliability, and efficiency (Rayson & Wilson, 2003).

2.3.6 Corpus Annotation

In corpus-based studies, grammars, semantic or pragmatic annotations are important (Mair et al., 2002). Nowadays, using off-the-shelf POS taggers, NERs, and lexical parser would be very beneficial. In corpus-driven annotation methods and concordances, wordforms is the elementary unit for analysis, and their frequency and statistical significance should be investigated and identified (Sinclair, 1991). Concordance tools such as Wordsmith detect statistically-significant collocates and perform cluster analyses (Scott, 1996).

2.3.7 Text Messages Corpus Compiling

The question of how to collect data is directly related to the nature of the data, and their approaches also change based on the nature of the data, and research questions (Hunston & Francis, 2000). The methods of data collection used for text messaging in the previous studies are generally suggesting the recruitment of family and friends (Segerstad, 2002). Recruitment of family and friends provides a great possibility of ensuring data authenticity (How & Kan, 2005; Segerstad, 2002). It also provides a better understanding of participants information and backgrounds and easy way to gather personal messages, and contents (Fairon & Paumier, 2006). However, automatic data collection has the advantages of saving time and cost (Kasesniemi & Rautiainen, 2002; Segerstad, 2002).

2.4 Discussion and Summary

This chapter discussed the distinction between corpus-based and corpus-driven language studies. Although, there are assumptions that corpus-based and corpus-driven approaches are fundamentally alike: both believe in corpus data and the significance of

frequently-occurring patterns. Corpus-based studies typically use corpus data to explore a theory or hypothesis, aiming to validate it, refute it or refine it, which is practiced in corpus linguistic studies.

Corpus-driven linguistic considers the corpus itself as the sole source of the hypotheses about languages. It is thus claimed that the corpus itself embodies a theory of language. In corpus-driven studies, pattern and statistical characterization of letters, words, and phrases would be investigated. Thus, the corpus-driven fitted the aim of this study (Section 1.4) because it is required to understand the pattern of letter repetition (e.g. "*hiiii*"), pattern of word abbreviations (e.g. "*wll dn*"), and frequency of unknown words. Similar to corpus-driven studies, this study does not test any pre-defined hypothesis. Chapter 4 discussed the result of our corpus-driven study.

The most important part of a corpus-driven study is the corpus building process. Corpus size, representativeness, transcript errors, and annotations should be considered. In this study, data are collected from Twitter. Twitter users are identified by their geolocation and language identification. Langid is an probabilistic open source Python library for languages detection (Lui & Baldwin, 2012). After utilizing Langid, our linguistic experts manually verify Twitter users who regularly send informal Tweets, in the Malay language from Malaysia. Then, 3200 public Tweets are automatically collected from each identified users via Twitter APIs. Chapter 4 illustrates our strategies for sampling methods, mixture of author styles, lifetime of text, and text storing/retrieving.

CHAPTER 3: TEXT NORMALIZATION

3.1 Introduction to Text Normalization

In real text, there are Out–of–Vocabularies such as numbers, abbreviations, dates, currency amounts and acronyms. Generally, OOVs could not be found in a dictionary. Text normalization is the process of converting non-standard words into their standard forms. The OOV words have a greater ambiguity than correct words in terms of pronunciation or interpretation (Sproat et al., 2001). In many software applications, input texts are normalized by converting the OOVs with the contextually appropriate correct word or sequence of words. In addition, the short text messages (SMS) and other instant messages are different from the normal written texts and have special characteristics such emoticon (Aw et al., 2006).

Instant messages such as SMS and Twitter contain shortened and non-standard forms of texts, simply because of the desire for rapid text entry. Furthermore, text messages are written in informal and even personal styles. These generate linguistic creativity, and many innovative lexical items (Cook & Stevenson, 2009; Thurlow & Brown, 2003).

Normalization is a problem that must be addressed before other NLP modules can take place. It is proven that normalization modules are beneficial in many applications such as summarization and information retrieval, sentiment analysis, and emotion detection (Pennell & Liu, 2011b; Sproat et al., 2001).

This chapter describes briefly the existing works on text normalization and related problems in noisy text, detecting, categorizing, cleansing, translating and converting.

3.2 Approaches in Text Normalization

The existing works are grouped into six different categories as shown in Table 3.1: noisy channel, Statistical Machine Translation (SMT), lexical, dictionary, classification, and hybrid. Choudhury et al. (2007) introduced a noisy channel using a word error model based on the Hidden Markov Model (HMM) approach. The HMM word error is modeled by a unigram language model that can normalize English SMSs with considerable accuracy (Choudhury et al., 2007).

Aw et al. (2006) suggested utilizing SMT methods to normalize noisy texts. In SMT approach, noisy text is considered as a source language, and the standard English as a target language. Word alignment between source and target text is the most challenging part of the SMT method in translation of two different languages. In contrast, the word alignment is not a crucial process in SMT methods for normalization, because usually an order of words is not disturbed in noisy texts such as Tweets (Aw et al., 2006).

In the lexical approach, after identifying OOV tokens and set of candidates are generated, the system calculates the dependency between tokens using the Stanford parser. The next step is to exploit a classifier to determine ill-formed words. The best candidate is selected using different metrics such as; phonemic edit distance, lexical edit distance, affix substring, Longest Common Subsequence (LCS), language model, and dependency-based frequency features.

The dictionary based approach is the easiest to understand approach. Specialized dictionary compiling is a usual practice in all domains of science and technologies. Conventionally, specialized dictionaries are made manually by language or domain experts. Automatic dictionary generation can help to reduce human efforts via extracting most suitable candidates, but in most of the research experiment still it is not fully automated (Raghunathan & Krawczyk, 2018).

The classification approach is more straightforward to use because it predicts the label of each token based on the morphological and contextual features. Finally, hybrid approaches are composed of rule-based components and trained models. The below sections describe the experiments on each approach in sequential order.

Table 3.1: Categories of Normalization Approaches

CATEGORY	Noisy Channel	SMT	Lexical	Dictionary	Classification	Hybrid
EXISTING WORKS	(Choudhury, et al. 2007)	(Aw, et al. 2006)	(Han & Baldwin, 2011)	(Clark & Araki, 2011)	(Mohanthy, et al. 2013)	(Beaufort, et al. 2010)
	(Cook & Stevenson, 2009)	(Schlippe, Zhu, Gebhardt, & Schultz, 2010)	(Wei, et al. 2011)	(Basri et al., 2012)	(Zhu, et al. 2007)	(Xue, Yin, & Davison, 2011)
	(Pennell & Liu, 2011b)	(Kaufmann & Kalita, 2010)	(Saloot, Idris, Shuib, Gopal Raj, & Aw, 2015)	(Samsudin, et al. 2012)	(Contractor, et al. 2010)	
	(Liu, Weng, Wang, & Liu, 2011)	(Pennell & Liu, 2011a)		(Han, et al. 2012)	(Pennell & Liu, 2010)	
		(Gadde, et al. 2011)		(Oliva, et al. 2013)	(Wang, et al. 2013)	
		(Li & Liu, 2012)				
		(Lopez et al. 2012)				
		(Wang & Ng, 2013)				
		(Ling, et al. 2013)				

3.2.1 Noisy Channel

Table 3.2 presents the four studies that used noisy channel. Choudhury et al. (2007) addressed the issue of text normalization and its problem in natural language processing. The authors conduct a thorough investigation on the text normalization. Firstly, the authors formalized of the entire problem; specifically, they define the

subtasks of the problem. Secondly, they proposed a unified approach to the whole task on the basis of tagging. Specifically, it takes the problem as that of assigning tags to the input texts, with a tag representing deletion, preservation, or replacement of a token. And then proposed a unified tagging approach to perform the task using Conditional Random Fields (CRF). It is proved that with the introduction of a small set of tags, most of the text normalization tasks can be performed within the approach. The accuracy of the proposed method is high, because the subtasks of normalization are interdependent which are performed together. Experimental results in the email normalization show that the proposed method significantly outperforms approaches that used cascaded models and independent models. The unified model can achieve better performances in text normalization, because the subtasks of text normalization are often interdependent. In addition, there is no need to define specialized models and features to conduct different types of cleansing; all the cleansing processes have been formalized and conducted as assignments of the three types of tags.

For the purpose of experimentation, Choudhury et al. (2007) used email data. Five thousand emails are randomly chosen in total from 12 newsgroups. The cascaded approach and the independent approach are used as comparison baselines. For the comparison baseline methods, several basic prediction subtasks such as; extra line break detection, extra space detection, extra punctuation mark detection, sentence boundary detection, unnecessary token detection and case restoration are defined. And CRF++ tool is used for implementing the method. In addition, based on experiments Choudhury, et al. (2007) approach significantly outperforms the two comparison baseline methods for text normalization. It can also be seen that the performance of the unified method decreases when removing the transitional features.

Cook and Stevenson (2009) analysed sample of creative, non-standard text message word forms in order to detect frequent word formation processes in texting language.

The stylistic variations and subsequence abbreviations formation types in this approach is approximately 66%, because the stylistic variations exhibit non-standard spelling, such as representing sounds phonetically. Also, the subsequence abbreviations are composed of graphemes in a standard form which are very frequent, but regularly omitting vowels. In respects to the frequent words, Cook and Stevenson (2009) construct an unsupervised noisy-channel model for text message normalization using a test set of 303 text message forms that differ from their standard form. The model they used was able to achieve an accuracy of about 59%.

Pennell and Liu, (2011b) describe a text normalization system for deletion-based abbreviations in informal text using statistical classifiers to learn the probability of deleting a given character using features based on character context, position in the word and the containing syllable. The approach is mainly used for generating multiple abbreviation hypotheses for a word, in order to ensure that the designed system is robust to different and previously unseen abbreviations.

Liu et al. (2011) attempt to model the generation process from the dictionary words to non-standard tokens using a sequence labelling framework. In this approach, each letter in the dictionary word can be retained, removed, or substituted by other characters. To avoid the expensive manual annotation process, the approach automatically collects a large set of noisy training pairs using a novel web based method and performed character-level alignment for model training. The authors conducted experiments using both Twitter and SMS messages and their results significantly outperformed the Jazzy spell checker.

Table 3.2: Comparison of Techniques Used in Noisy Channel

No.	Paper	Techniques	Dataset	Accuracy	Shortcoming
1.	(Choudhury, et al. 2007)	word level HMM model + character level HMM model (grapheme + morpheme + cross-linking)	1000 English parallel (20,000 words) SMS for training + 1228 unseen tokens for testing	89% (accuracy)	Does not consider transposition errors; supervised model.
2.	(Cook & Stevenson, 2009)	Unsupervised version of noisy channel using two categorizations: Stylistic variation & Subsequence abbreviations	Dataset of Choudhury et al. (2007)	59.4% (Accuracy)	Word formation categories have over lapping and ambiguity.
3.	(Pennell & Liu, 2011b)	Abbreviation modelling based on annotated data and predefined categories.	705 Tweets annotated with at least one deletion-based abbreviation	223 (Top1), 289 (Top10)	Address only a portion of the normalization problem: abbreviation
4.	(Liu et al., 2011)	Automatically creating training set by Google queries.	Training data: English part of Edinburgh Twitter corpus. Testing data: 303 parallel OOV (from SMS) + 3,802 parallel OOV (from Twitter)	Twitter: 59.15% (Accuracy) SMS: 58.09% (Accuracy) Combined with conventional Java spell checker: Twitter: 68.88% (Accuracy) SMS 62.05% (Accuracy)	Does not handle acronyms (one-to-many).

3.2.2 Statistical Machine Translation

Table 3.3 presents the comparisons of techniques for SMT metaphor. Aw et al. (2006) proposed a phrase-based statistical MT model for the task. It is an adapted statistical machine translation model. The authors viewed SMS normalization as a translation problem from the SMS language to the English language. SMS normalization is treated as an MT problem where the SMS language is to be converted to normal English. The model was evaluated using 5-fold cross validation on a parallel SMS normalized corpus with 5000 sentences. The result shows that the method can achieve 0.80702 in BLEU score against the comparison baseline BLEU score 0.6958.

Schlippe et al. (2010) implemented an SMT-based French text normalization system with a web-based interface that provides training material in the form of parallel corpus, for both normalized and non-normalized text from native speakers. Gadde et al. (2011) proposed an algorithm for generating noise across various texts. The algorithm generates noisy-regular parallel data for generating training data for MT models.

Kaufmann and Kalita (2010) proved that by combining statistical SMT and preprocessor, removing the majority of noise from Tweets is possible. Tweets have been preprocessed to remove much noise as possible and then feed them into an SMT model to convert them into standard English. In addition, Lopez Ludeña, San Segundo, Montero, Barra Chicote, and Lorenzo (2012) proposed an architecture that composes of three modules, which are; a tokenizer module, a phrase-based translation module, and a post processing module for removing extra tokens.

Pennell & Liu (2011a) described a two-phases approach for SMS normalization. The first phase uses a character-level MT system to generate possible hypotheses for each abbreviation. While the second uses a language model (LM) to choose the hypothesis

in context. This model is quite different from other models in such a way that, the MT model is trained at the character-level during the first phase; that is, instead of learning mappings between words and phrases, character maps to another character. The mapping can be in the form of many-to-many mapping. Li and Liu (2012) proposed an approach to segment words into blocks of characters according to their phonetic symbols and apply MT and sequence labeling models on the block-level.

Wang and Ng (2013) developed a novel beam-search decoder for normalization of social media text. The decoders for text normalization effectively integrates multiple normalization operations, among are; firstly, a punctuation correction method based on a Dynamic Conditional Random Field (DCRF) models that produce normalized candidates. Then, a missing word recovery based on a CRF mode that produces another results for the beam decoder. Ling et al. (2013) introduced another data-driven approach for microblog normalization. The authors developed a corpus of Tweets. Then, two models (i.e., The phrase level and character level models) that learn generalizations of the normalization process, and a decoder that combines both models during decoding were also developed.

Table 3.3: Comparison of Techniques Used in SMT

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
1.	(Aw, et al. 2006)	Phrase based SMT (EM algorithm and Viterbi)	5000 English parallel SMS messages, which consists of raw (un-normalized) SMS messages and reference messages	0.8070 (BLEU)	0.5784 (BLEU)	Highly depends on training set. Does not consider punctuations.
2.	(Schli	web-based	4000	94.4		User's

Table 3.3: Continued

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
	ppe et al., 2010)	interface to collect training data+ SMT-base normalization	French sentences for training crawled from French online newspapers. + 1000 sentences for testing	(BLEU), 3.9 (Levenshtein), 471.0 (perplexity)		mistakes on normalization specifically on numbers. Cannot normalize unseen data.
3.	(Kaufmann & Kalita, 2010)	Orthographic Normalization + Syntactic Disambiguation + SMT-like normalization	1150 parallel Tweets.	0.7985(BLEU) 11.7095 (NIST)	0.6799 (BLEU) 10.5693 (NIST)	Does not consider phonetics.
4.	(Pennell & Liu, 2011a)	Character-level SMT	Training & testing: Manually annotate 4661 Tweets that sent via SMS.	60.39 (Top1), 74.58 (Top3), 75.57 (Top10), 75.57 (Top20)	Normalized by Jazzy: 49.86 (Top1), 53.13 (Top3),54.78 (Top10) 55.44 (Top20)	Does not handle acronyms (one-to-many).
5.	(Gadde, et al. 2011)	Artificially generating noisy text to train SMT-like normalization system.	NUS corpus as a language model. WSJ and BNC as a input to noise generation system. Aw et al., 2006, Choudhury et al., 2007 and Contractor et al., 2010 as a test set	Aw et al., 2006 test set: 0.531 (BLEU) Choudhury et al., 2007 test set: 0.482 (BLEU) Contractor et al., 2010 test set: 0.503 (BLEU)	Aw et al., 2006 test set 0.448(BLEU) Choudhury et al., 2007 test set : 0.396 (BLEU). Contractor et al., 2010 test set: 0.410 (BLEU)	Low BLEU score compared to other normalization systems.
6.	(Li & Liu, 2012)	Character-block Level Machine Translation + Character-	SMS dataset: 303 pairs of words from (Choudhury	SMS: 74.6 (Top1), 84.6 (Top3), 90.3 (Top10), 92	Not reported	Still training size is considerable.

Table 3.3: Continued

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
		Block Level Sequence Labelling + Character-level Two-step MT + Jazzy Spell Checker	et al., 2007). Twitter dataset: 3,998 pairs words from 6160 Tweets of (Liu et al., 2011).	(Top20). Twitter: 62.6 (Top1), 75.1 (Top3), 84 (Top10), 87.5 (Top20)		
7.	(Lopez et al. 2012)	Tokenization (rules) + identify NSW+ Phrase based SMT+ Detokenization	5800 numbers+ 5225 abbreviated sentences	Numbers: 97.5 (BLEU) 1.5 (WER) Abbreviations: 96.1 (BLEU) 2.5 (WER)	Not reported	Highly depends on training set.
8.	(Wang & Ng, 2013)	Punctuation Correction + Missing Word Recovery + Beam-search decoder SMT	2,000 SMS from the NUS English corpus normalized into formal English and then translated into formal Chinese.	Intrinsic: 66.54 (BLEU) Extrinsic: English-to-Chinese translation: 22.81 (BLEU)	Intrinsic: 37.38(BLEU) Extrinsic: English-to-Chinese translation: 13.63(BLEU)	Language translation cause to eliminate lexical feature of the original text
9.	(Ling, et al. 2013)	pharse-based SMT + Character-based SMT	1.3M parallel Tweets for training. 1290 English-Mandarin microblog sentence pairs for development. 1291 English-Mandarin microblog sentence pairs for testing.	Intrinsic: 22.91 (BLEU). Extrinsic English to Mandarin :15.94 (BLEU)	Intrinsic: 19.90 (BLEU). Extrinsic English to Mandarin :15.10 (BLEU)	Highly depends on training set.

3.2.3 Dictionary

Table 3.4 provides the four dictionary based normalization methods. Clark and Araki (2011) introduced manually developed database with 1,043 entries using a trie-type data structure. The trie-type data structure is a type of data structure that accommodates unlimited word length (i.e., Both single words and phrases). Basri et al. (2012) proposed a new Malay spell checker that detects and automatically corrects misspelled words in Malay without any interaction from the user. The proposed approach automatically replaces the misspelled word if it exists in the dictionary, otherwise, it will go through the process of rule-based normalization. However, if the words cannot be identified as misspelled words, few alternative words will be suggested and ranked using the Levenshtein distance.

In addition, Samsudin et al. (2012) stated that a dictionary of translation of 5000 Malay noisy terms was successful in normalizing micro-texts. Also, Oliva, Serrano, Del Castillo, and Igesias (2013) introduced a special Spanish phonetic dictionary to address the shortcomings of the Spanish dictionary approach, in which each entry is formed by a coded consonant string, vowel strings, and their positions in the word, for normalizing Spanish SMS texts.

Table 3.4: Comparison of Techniques Used in Dictionary

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
1.	(Clark & Araki, 2011)	Casual word context-aware dictionary (1,043 entries: English words & phrases)	100 sentences from Twitter for testing	Extrinsic evaluation: (3.34 Japanese MT error)	Extrinsic evaluation: (3.34 Japanese MT error)	Highly depends on number of dictionary entries.
2.	(Basri et al., 2012)	Tokenization+ +eliminating symbols + eliminating English word + Stemming + Dictionary loo-	Tested against 11896 words (3046 distinct words)	*91.22%* (accuracy)		Eliminating English words and symbols causes data lost. Stemming words causes data lost and

Table 3.4: Continued

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
		up + reduplication normalization + Selangor pronunciation normalization + Negative word normalization	from Malay weblogs.			inaccurate normalization. Depend on the size of dictionary. Language dependent.
3.	(Samsudin, et al. 2012)	Collecting data+ manually normalizing NSW to build a dictionary	1000 positive and 1000 negative online movie reviews, created by Malaysians.	Extrinsic evaluation: 66.60 (K-nearest), 85.55 (Naïve Bayes), 82.85 (SVM)	Extrinsic evaluation: 62.35(K-nearest), 80.95 (Naïve-Bayes), 79.55(SVM)	It is not context-aware dictionary. It depends on the size of dictionary.
4.	(Oliva, et al. 2013)	Tokenization + Spanish SMS dictionary + Spanish phonetic dictionary+ disambiguation	92 Spanish SMS + a bilingual book (Spanish –SMS language)	0.8054 (BLEU) 0.1381 (WER)	0.12 (BLEU)	Not successful with words with very few characters and also proper nouns.

3.2.4 Lexical

Table 3.5 presents three studies that use lexical approaches. Han and Baldwin (2011) proposed a lexical normalization Twitter and SMS data based on morphophonemic variation for most ill- formed words. After detecting ill-formed words via classifiers, the proposed approach generates the correct candidates based on morphophonemic similarity. The approach uses lexical edit distance, phonemic edit distance, prefix substring, suffix substring, and the Longest Common Subsequence (LCS) to select the best candidate. Wei et al. (2011) employed a lexical normalizer as a preprocessing step for time sensitive Twitter search. In addition, the lexical approach applied in

Singaporean English by (Saloot et al., 2015) using maximum entropy for candidate selection.

Table 3.5: Comparison of Techniques used in lexical

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
1.	(Han & Baldwin , 2011)	Candidate generation + Candidate Selection	Testing: 549 English Tweets + Choudhury et al., 2007 SMS corpus	SMS: 0.756 (precision), 0.754 (Recall), 0.755 (F-score),: 0.876 (BLEU) Twitter: 0.753 (precision), 0.753 (Recall), 0.753(F-score), 0.934 (BLEU)		Cannot normalize acronyms; in extreme highly noisy text, no useful context feature can be extracted. Steps are linearly combined, while a weighted combination is likely better which need tuning effort
2.	(Wei, et al. 2011)	Lexical normalization + Time sensitive query on Twitter	6 milion Tweets downloaded from TREC Microblog dataset	0.2220(R-Prec) 0.1734(MAP)	0.2191 (R-Prec), 0.1710 (MAP)	normalization itself causes many noises.
3.	(Saloot et al., 2015)	Lexical normalization + maximum entropy candidate selection	7,000 parallel Singapore an English Tweets	83.12 BLEU score	42.01 BLEU score	No usage of text semantic similarity features

3.2.5 Classification

Table 3.6 lists five classification methods for text normalizers. Zhu et al. (2007) proposed a unified tagging approach using CRF++. This approach specifically treats text normalization by assigning tags to represent either deletion/preservation, or replacement of tokens in the text. Contractor et al. (2010) uses an unsupervised method

to proposed two steps given noisy sentence. A weighted list of possible clean tokens for each noisy token are obtained with this approach. The normalized sentence is then obtained by maximizing the product of the weighted lists and the language model scores.

Pennell and Liu (2010) described a normalization system for text messages that allow them to be read by a text-to-speech (TTS) engine. To address the large number of texting abbreviations, statistical classifier is used to learn when to delete a character. These features are based on character context, function and position in the word and containing syllable. To ensure that the system is robust to different abbreviations for a word, multiple abbreviation hypotheses are generated for each word based on the classifier's prediction.

Table 3.6: Comparison of Techniques Used in Classification

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
1.	(Zhu, et al. 2007)	4,168,723 features were used in a CRF model	5000 parallel English emails	86.46 (Precision) 93.92 (recall) 90.04 (F-score) 99.05 (accuracy)		Costly annotated data.
2.	(Contractor, et al. 2010)	Noisy tokens to clean tokens mapping + Rule-based tokenization + Language model and noisy channel	800 SMS from NUS corpus	48 (BLEU) 30 (WER)	40 (BLEU) 40 (WER)	Not achieved significant accuracy score.

Table 3.6: Continued

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
3.	(Pennell & Liu, 2010)	Rule-based translate English to texting lingo + Automatic abbreviation creation + Mapping abbreviations to English words	20,000 English SMS	58.46 (Top1), 69.23 (Top2), 75.38 (Top3)		Address only a portion of the normalization problem: abbreviation
4.	(Wang, et al. 2013)	Word segmentation+ OOV Recognition + candidates generation + + Rule-based feature extraction + Statistical features extraction + Supervised classification	1036 unique word pairs	0.886 (precision), .443 (recall), 0.590 (F-score)		Supervised method.
5.	(Mohanthy, et al. 2013)	Rule-based pipeline architecture	No Experiment reported.	No Experiment reported.	No Experiment reported.	No Experiment reported.

Wang et al. (2013) formalized task as a classification problem and proposed rule-based and statistical features that explain the connection between formal and informal pairs. The two-stage selection-classification model is evaluated on a crowdsourced corpus and achieved a normalization precision of approximately 89.5%. Mohanthy et al. (2013) defined the normalization problem as a task of noise elimination and boundary detection subtasks. Although, they proposed an approach that specifically treats text normalization by assigning tags representing deletion, preservation, or replacement of the tokens in the text.

3.2.6 Hybrid

Beaufort et al. (2010) is introduced an architecture to normalize French text messages using noisy channel, SMT, and lexical similarity. The architecture was developed using three modules in a pipeline. The first module accomplished tokenization with a set of hand written rules. The second module normalizes the tokens based upon the trained phonetic model. And the last modules accomplished de-tokenization on a set of hand written rules. Although, significant accuracy was obtained in terms of BLEU and WER scores. However, the significant accuracy of SER is not encouraging because of the phonetic complexity in French language (Beaufort et al., 2010).

Automatic Speech Recognition (ASR) was combined with a noisy channel approach to normalize Twitter messages by (Xue et al., 2011). The authors used a channel probability equation to integrate grapheme channel, phoneme channel, context channel, and acronym channel. Although the results show higher recall than other methods and algorithms, it produces low precision because of high error rates on normalizing proper nouns. The comparison of techniques used in hybrid approach is presented in Table 3.7.

Table 3.7: Comparison of Techniques Used in Hybrid

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
1.	(Beaufort, et al. 2010)	Tokenization(rules)+ Identify NSW words (lexical look up) combine (noisy channel metaphore, SMT) + de-tokenization	30,000 French SMS (character level aligned) + A lexicon of 430,000 inflected forms derived from Morlex	0.83 (BLEU) 9.31 (WER) 65.70 (SER)	0.47 (BLEU)	Does not consider phonetic similarities. High complexity of the Model.

Table 3.7: Continued

No.	Paper	Techniques	Dataset	Accuracy	Baseline	Shortcoming
2.	(Xue et al., 2011)	grapheme channel + phoneme channel + context channel + acronym channel	Testing & training set: 818 parallel Tweets + Choudhury et al., 2007 SMS corpus	Twitter: 0.96 (Accuracy) (0.61) F-measure 0.76 (Precision) 0.79 (Recall) SMS: 0.96 (Accuracy) (0.91) F-measure 0.93 (Precision) 0.90 (Recall)	(Aspell) Twitter: 0.92 (Accuracy) (0.06) F-measure 0.08 (Precision) 0.05 (Recall) SMS: 0.63 (Accuracy) (0.13) F-measure 0.23 (Precision) 0.09 (Recall)	Supervised approach.

3.3 Malay Text Normalization

To the best of our knowledge, there are two published studies on Malay normalization. Firstly, Samsudin, Puteh, Hamdan, and Nazri (2012) developed a dictionary-based system known as NoisyTerm which normalize the content of Malaysian online media. NoisyTerm converts words or tokens to the standard Malay disregard of their context. Therefore, when a token has multiple translations, there would be an ambiguity to select the best translation. Secondly, Basri, Alfred, and On (2012) introduced a Malay normalization approach that has not evaluated by any standard metrics yet. In Basri et al. (2012), a Malay stemmer is used (Kadir, Musa, Azman, & Abdullah, 2011). Kadir et al. (2011)(Basri et al., 2012)(Basri et al., 2012) stemmer was designed to work in standard Malay. Using the stemmer caused the loss of the original terms, while original words need to be normalized along with their affixes.

3.4 Summary

Table 3.8 summarizes the merits and demerits of the normalization approaches. The first text normalization task has been handled through the noisy channel. Since, the

noisy channel is usually used in spelling correction tasks, it is also known as spell correction metaphor. The noisy channel approach addresses the normalization task on a word-per-word basis by employing a supervised noisy channel model. Choudhury et al. (2007) introduces hidden Markov model using manually annotated training data. However, unlike the SMT-like system, the model discounts the context around the token.

Table 3.8: Summary of Normalization Approaches

	Noisy channel	**SMT**	**Lexical**	**Dictionary**	**Classification**	**Hybrid**
Merit	Very similar to spell checkers	High precision	Combining different text similarity modules	Simple design	Simple implementation	Inheriting the merits of combined approaches
Demerit	No solution for one-to-many abbreviations	Cannot normalize unseen data; low recall	Need tuning effort between candidate generation methods	Highly depends on number of dictionary entries.	Costly annotated data	Inheriting the demerits of combined approaches

Aw et al. (2006) attempted to employ the SMT method for normalization. The texting language is considered as a source language, and the standard English as a target language. The main shortcoming of this approach is that the system can only normalize tokens that are existing in the training set. The SMT approach has since been re-examined, expanded and improved by other researchers (Lopez Ludeña et al., 2012). For example, Kaufmann and Kalita (2010) used the SMT-like approach to normalize English Tweets.

Han and Baldwin (2011) introduced a novel lexical approach for normalization. In the lexical approaches, after discovering the ill-formed words, corrections are generated based on the morphophonemic similarity. The most appropriate candidate is found using several measures such as; phonemic and lexical edit distance, Longest Common Subsequence (LCS), affix substring, dependency-based frequency, semantic similarity,

and LM. The approach needs tuning effort between the candidate generation methods, and functions poorly in highly noisy texts.

The simplest text normalization approach is the dictionary based approach approaches (Clark & Araki, 2011; Saloot, Idris, & Mahmud, 2014). This approach requires a dictionary whose entries are OOV and standard form pairs. It has been proven that using a colloquial dictionary can outperform other complex approaches such as statistical and lexical approaches (Clark & Araki, 2011; Saloot, Idris, & Mahmud, 2014). However, its performance depends on the size of the dictionary.

There are several classification methods which are successful to be applied to this task. The classification is the conventional approach in most of NLP tasks. Mohanthy et al. (2013) uses enhanced Named Entity Recognitions (NERs) for the informal text normalization task. The classification method also needs annotated training data similar to the SMT approach. Finally, merging the two approaches produce a third approach, which aims to avoid the shortcomings of the original approaches. For example, Beaufort et al. (2010) achieved high performances (i.e. About 9.3% Word Error Rate and 0.83 BLEU score) while merging the SMT-like and the spell checking approaches to normalize French SMSs.

There is a similarity between this research and the previous work by Basri et al. (2012), which is the usage of the characteristics of Malay Internet lingo writing style. However, this research represents a more comprehensive analysis on the Malay Internet lingo writing style than Basri et al. (2012). We normalized the noisy terms in a different manner compared to Basri et al. (2012) and NoisyTerm (Samsuddin et al., 2012). In this research, no stemmer is used to avoid loss of the characteristics of the original terms such as affixes. This research also addressed the ambiguity problem as occurred in NoisyTerm (refer to Section 3.3) using a context-aware approach. In context-aware approach, a best candidate will be selected from dictionary entries based on the

surrounding words in the sentence. Details of our context-aware approach will be discussed in Section 5.2.

CHAPTER 4: MALAY LANGUAGE TWEET CORPUS-DRIVEN ANALYSIS

4.1 Introduction

The design of the proposed normalizer depends on the results of the corpus-driven analyses. Chapter 2 discussed the detail corpus-driven studies. Corpus-driven studies do not have any pre-assumed hypothesis. A corpus-driven experiment extract features of a language which is represented by a corpus. There is a difference between corpora and datasets. Corpora should be comprehensive enough to represent a language or a subset of a language. While there are several Twitter datasets, Twitter corpora should be designed and evaluated in order to represent the Twitter informal language. This chapter discusses our corpus-driven experiment which is divided into two different sections. In the first section, a Twitter corpus is compiled due to the unavailability of a Malay language Twitter Corpus. Corpus is the compilation which is different from dataset gathering, where in corpus compilation, it should follow corpus design criteria such as variety and representativeness. The second section provides our contribution to the corpus analysis.

4.2 Malay Chat-style-text Corpus (MCC)

This section presents the process of compiling Malay Chat-style-text Corpus (MCC), concordance program and corpus structure. Sample selection has been done based on predicting sampling standards, variety, ethical issue, and chronology. The concordance undertakes corpus expandability and annotatability. The specific structure of the corpus helps concordance to interrogate the corpus in several ways.

The MCC, which contain about 1 million Tweets is made up of 14,484,384 word instances, 646,807 unique vocabularies, and metadata, such as used Twitter client application, posting time, and type of Twitter message (Tweet, Retweet, Reply).

4.2.1 Corpus Compiling

Sampling theories prove that sampling techniques have more priority than size in achieving representativeness (refer to Section 2.3.3). Compiling is divided into four stages:

1. In the first stage, a target population is defined. Defining an explicit target population helps to evaluate the representativeness of the corpus. The boundary of our desired population is provided in Figure 4.1. The figure refers to Malay language chat-style text.

2. The second stage is to declare a sampling frame. A sampling frame refers a list of potential sources for the target population. For example, if the target population is "*present engineering English lingo"*, the sampling frame should be a large number of engineering magazines, books, journals, and brochures that published recently in English language. In our case, sampling frame includes 4500 Twitter users-ID that belongs to users who set Malaysia as his or her location. These numbers of user-IDs are fetched via Twitter third party applications and Twitter APIs. In the next stage this 4500 user-Ids will be shortlisted based on a sampling technique.

3. The third stage deals with selecting a sampling technique and narrowing down the sampling frame. There are two major sampling approaches, which are; the probability and non-probability approach (refer to Section 2.3.3). In non-probability sampling, some elements are considered as out-of-coverage in the sample frame (refer to Section 2.3.3). Choudhury et al. (2010) proves that pure context-based sampling techniques such as location-based perform better than probability-based sampling in Twitter corpus information diffusion. Purposive sampling is a type of non-probability sampling that function based

on the characteristics of each source in the sampling frame (refer to Section 2.3.3). Therefore, to perform a precise selection of sources, purposive sampling scheme has been chosen in this study. For example, in the sample frame, there are users who post their Tweets in English language, thus they should be considered as out-of-coverage. Since the population is a representation of chat-style text, MCC should only encompass informal or production text (refer to Section 2.3.3). In this perspective, Tweet posts in a formal Malay language, such as commercial and political messages must be avoided. To do so, a native linguistic expert is employed to investigate the sampling frame and we found 321 users who post their messages in chat-style Malay language.

4. The last stage deals with gathering data according to the selected sampling technique. In purposive sampling, equal number of samples gathered from each source in the sampling frame. A computer application is then developed via Twitter APIs to extract 3200 messages from each user as shown in Figure 4.2. Although Twitter applied some restrictions on number of automatic fetched Tweets, our application can fetch maximally 3200 Tweets per user for a specific period of time.

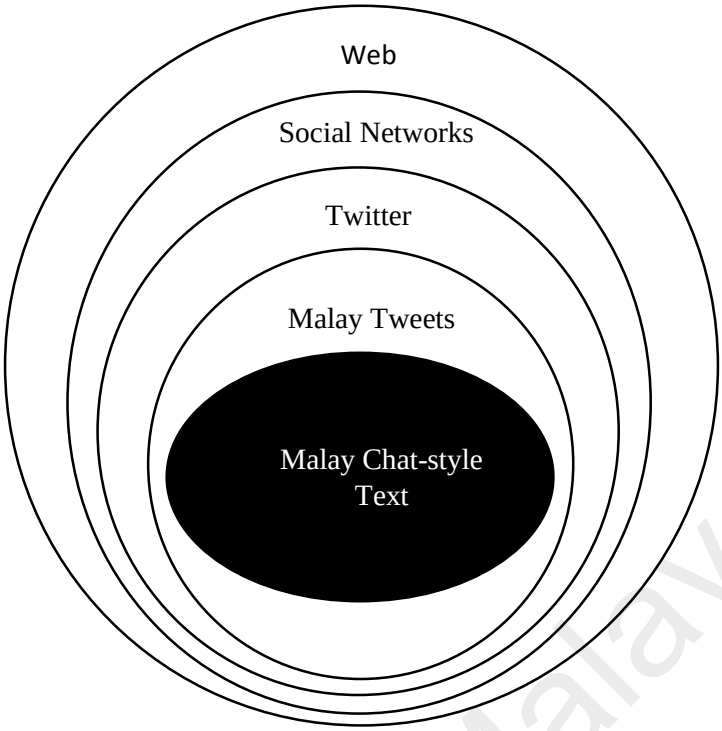

Figure 4.1: Filled Area Refers to Population

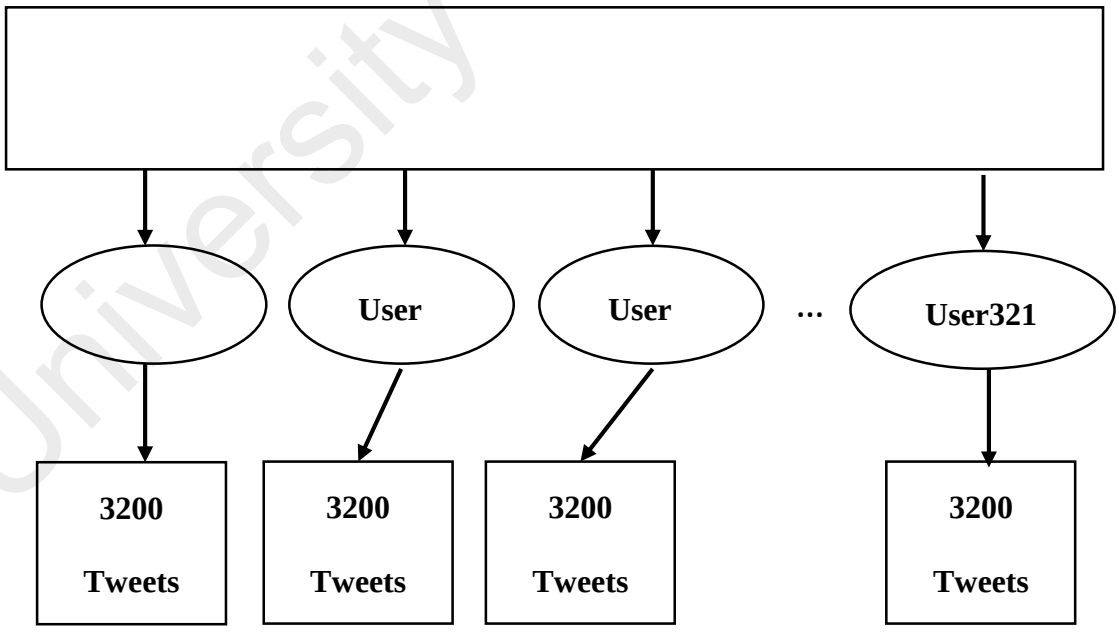

Figure 4.2: Sampled Data

4.2.2 Variety and Chronology

Section 2.3.4 explains the variety of language styles based on different usage of Lexis, grammar and discourse features between authors. Therefore, we selected more than three hundred users to cover a variety of styles. Originally, MCC is a synchronic corpus because gathered messages belong to a particular time, making the aim of this project a contemporary study of Malay chat-style language. The compilation process began in November 2012, when the application began extracting the earliest Tweet until it reached the 3,200th Tweet which is considered the latest message published by a user. However, it must be noted that the date of the latest Tweet differs amongst users because the number of posted Tweets per day is different between them. To have a synchronic corpus, our native linguistic expert selects users who at least disseminate 70 Tweets per week. Consequently, after corpus compilation, Figure 4.3 represents the number of collected Tweets per month.

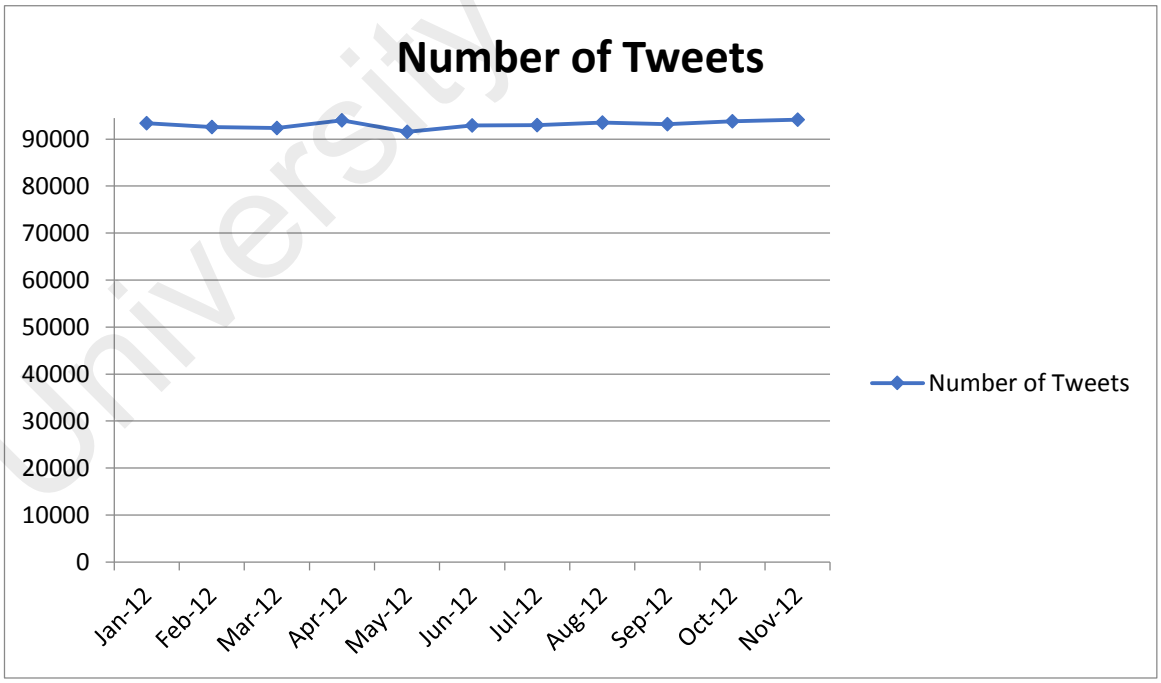

Figure 4.3: Time Frame of MCC

4.2.3 Ethical and Legal Issues

With the expansion of corpus linguistic, anonymizing has become one of the most significant parts of corpus legalization (Hasund, 1998). Apart from selecting user with a public profile, anonymizing can be done through converting user-IDs into fixed-length digest value. MD5 (Rivest, 1992) hash algorithm can produce fixed-length digest values.

Twitter user-IDs are decimal digits with arbitrary length between 8 or 9 decimal numbers. To avoid collision between user-IDs, the anonymizing process has been accomplished by applying MD5 (Rivest, 1992) hash algorithm on user-IDs. As shown in Figure 4.4, MD5 converts 8 or 9 decimal digits (i.e. Twitter User Ids) into a sequence of 32 digits hexadecimal number.

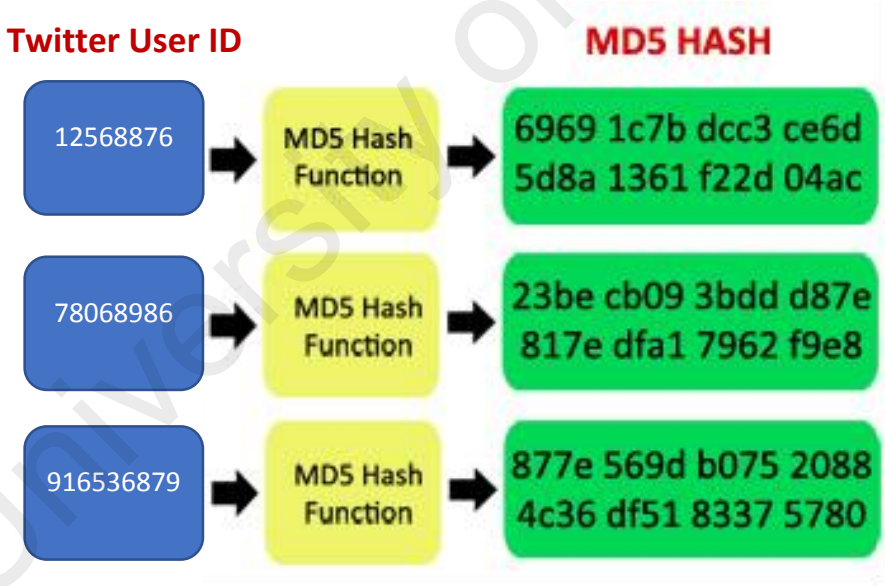

Figure 4.4: Anonymization using MD5

In addition, MCC is considered to be a free available corpus for research purpose and not for commercial purpose. Therefore, the corpus is distributed under a creative commons license. The license allows the general public to share, modify and review the corpus.

4.2.4 The Corpus Structure

The corpus has two different versions that is, the text and XML versions. The text version is a simple text file that is made up of only discourses. The text version can be used through most of linguistic concordance programs such as WordSmith (Scott, 1996) and also through the specific designed concordance for the corpus. XML version contains 1 million Tweets and their meta-data. The metadata of each Tweet includes Tweet-ID, user-ID, sending time, type of Tweet, and Twitter client application. The XML version schema is a series of Tweets. Each Tweet contains the text and its associated meta-data. The concordance program can retrieve information based on how the corpus is arranged irrespective of time. Figure 4.5 represents the illustration of corpus structure.

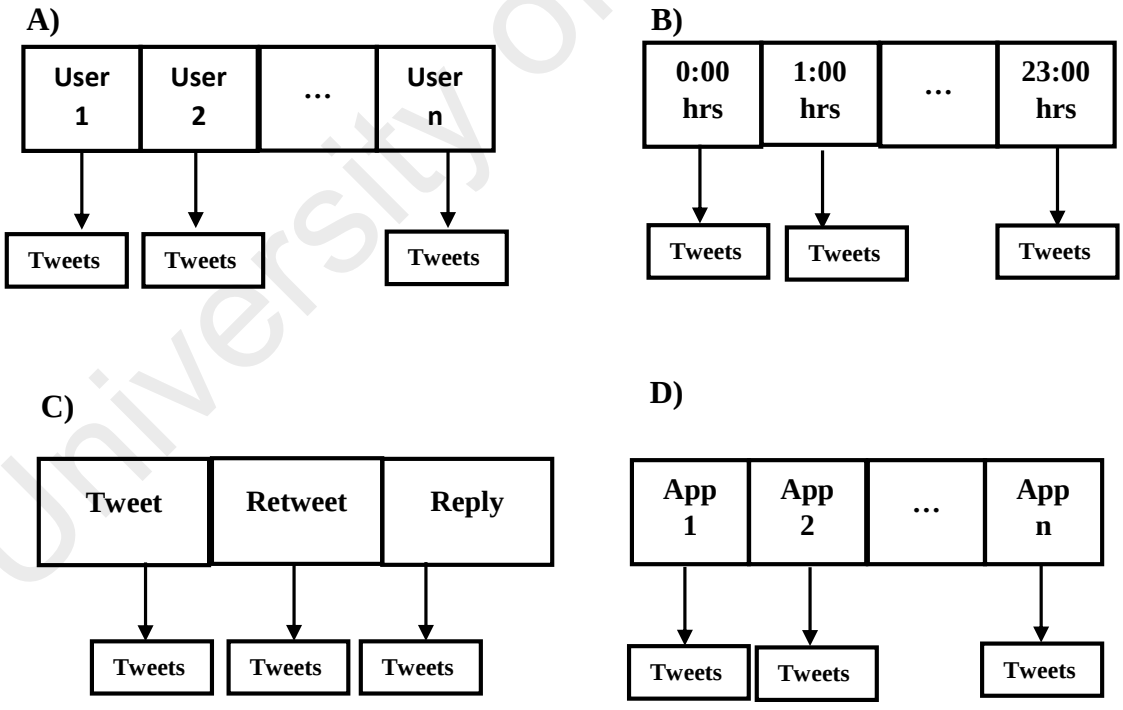

Figure 4.5: Corpus Structure

Figure 4.5 shows the structure of the corpus. Figure 4.5- (a) indicates that users can inspect Tweets according to a selected user-ID (User-ID is 32 digits unique hexadecimal number). Figure 4.5- (b) shows that users can inspect Tweets according to sending time (sending time is a Coordinated Universal Time (UTC) +0000). Figure 4.5- (c) indicates that users can inspect Tweets according to the desired type of Tweet. Twitter has three types, namely: 1) simple Tweet, a short text message limited to only 140 characters sent by a user; 2) Retweet, taking a Tweet from another user and resending the same Tweet; 3) Reply, a Tweet that looks like commenting about or responding to another Tweet. Figure 4.5- (d) indicates that users can inspect Tweets according to the demanded Twitter application. Twitter is not only a Website but it is a Web service that provides microblogging facilities to its users. This means, users can use Twitter without forcing them to use Twitter only through the Website. Twitter can be used in different ways like in Short Message Service (SMS), smartphone Twitter applications, and Personal Computer (PC) Twitter applications.

4.2.4.1 Annotation

To compile a parallel dataset, the MCC corpus is partially annotated with standard Malay translation. Two linguists from University of Malaya have converted 9,000 Tweets from MCC to their corresponding normalized Tweets that encompasses 33,878 OOV instances. The linguists manually align the noisy Tweets and their corresponding standard Tweets by Tweet and word. An interactive GUI is designed to assist the mapping procedure by the linguists. The conversion (i.e. translation) and mapping process took a total of 500 work hours. 7500 pairs are used in the implementation of the normalizer architecture by converting them to a colloquial dictionary (refer to Section 5.2.3), and another 1500 Tweets are used for testing the architecture (refer to Section 6.3.2).

4.2.5 Concordance Program

An exclusive concordance program was written in the Python programming language and placed in the corpus package. It is an open source program with a Graphical User Interface (GUI), which is capable of doing an ordinary concordance task as shown in Figure 4.6. In particular, the program manages to list the most frequent words, and the most frequent collocations, search a particular word to state its frequency and display Tweets with that word. However, one of the most significant features of this program is the ability to work with the corpus meta-data (User-ID, application, type of Tweet, and hour). The program can list out all existing user-IDs and all used Twitter applications, thus, users can select any and view related Tweets. Simple Tweet, Reply, and Retweet are three genres that users can select to observe only related Tweets to that genre. Finally, the program can also display Tweets that have been posted in a particular hour.

Figure 4.6: Concordance User Interface

Another important feature of this program is the ability to add new Tweets into the corpus. Users only need to enter a Twitter public user-ID, the program then fetches 3,200 latest Tweets from it, and add the new Tweets to the corpus. It should be noted that MCC corpus is not an annotated corpus and the designed program is not doing any annotating task, such a Part of Speech (POS) tagging. However, the concordance provides a convenient way for human annotator to annotate the corpus. That is, the

concordance can manipulate the XML file to add new tags and associate the tags to tokens.

4.2.6 Evaluation

4.2.6.1 Cartography

We draw geolocation of the Tweets on a world map in order to track the actual place of users while sending messages. Most of the Twitter users do not allow their ubiquitous devices to send geolocation data together with their Tweets due to concerns about privacy. Out of one million gathered Tweets, only less than 22,000 of them have geolocation data. To compare the dispersal patterns in MCC with the current Malaysia population density, we use a population density map of Malaysia, as shown in Figure 4.7. The population of Malaysia at the time of this study is about 28 million, in which the Federal Territory of Kuala Lumpur and Johor are the first and second most populated states. A latitude/longitude coordinates of the MCC posts drawing is displayed in Figure 4.8, where Figure 4.8- (a) shows that the distribution of the origins are concentrated in Malaysia, and Figure 4.8- (b) shows that the distribution of the origins are concentrated in the Federal Territory of Kuala Lumpur and Johor. Accordingly, we can find many jargons and slangs in the corpus that belongs to the KL dialect. Therefore, there is a similar distribution pattern across the geolocations of the MCC posts and population density of Malaysia that confirms the sampling frame is represented the population.

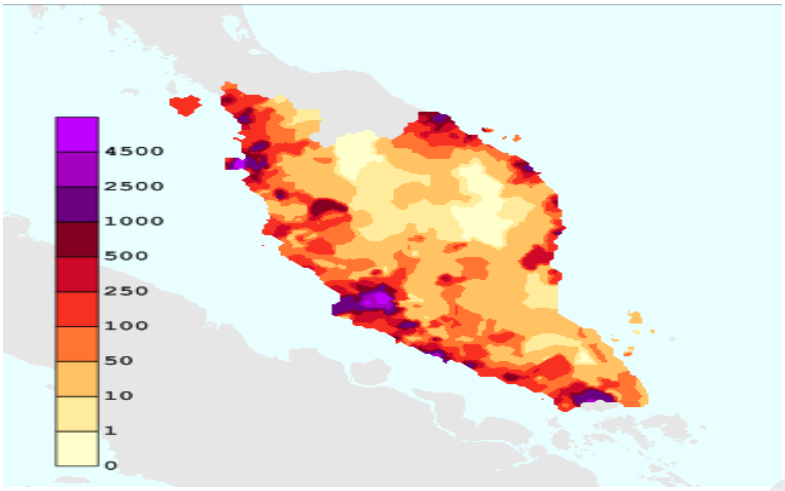

Figure 4.7: Malaysia population density map

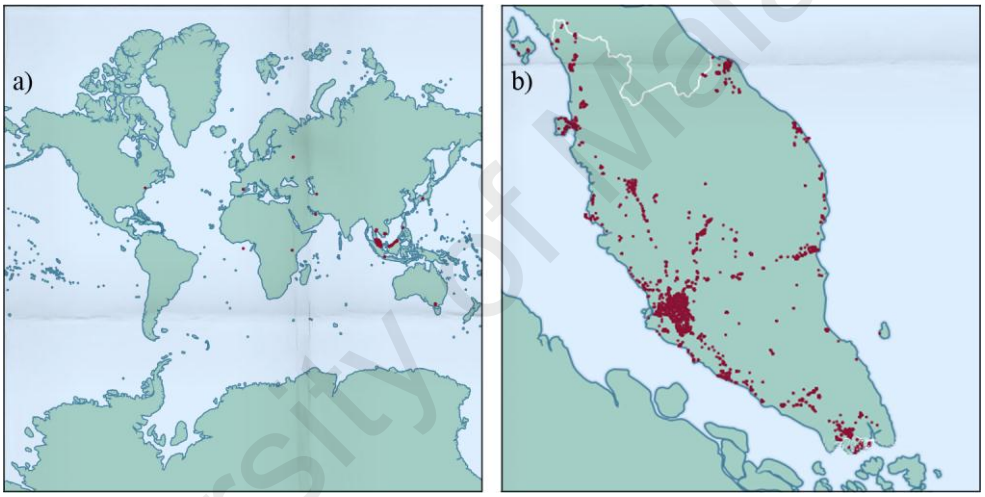

Figure 4.8: Geolocation of MCC nodes

4.2.6.2 Language Identification

In the process of corpus compiling, an expert selects Malaysian Twitter users. Using an automatic language identification tool helps to evaluate the representativeness of this selection, and, thus, to evaluate the representativeness of the corpus. We utilize Langid.py (Lui & Baldwin, 2012), which is a state-of-the-art language identification Python library. The results of language identification are presented in Table 4.1. Langid.py implements a normalization on log-probability that produces a confidence score between [0, 1]: 1 refers to complete confidence and 0 to a failed identification.

The language identification process has been done in only one step. The Langid applies to the text version of the corpus (a single text file). As shown in Table 4.1, the Malay language is identified with a high confidence value and English is detected with low confidence. Unsurprisingly, French is not detected in the corpus. Overall, the confidence scores show that the boundary of the desired population (Malay chatting people) is well determined.

Table 4.1: Language identification results

ISO 639-1 language code	Language name	Confidence score
ms	*Malay*	0.896170834
en	*English*	0.212789086
fr	*French*	Not

4.2.6.3 Zipf's Law

Zipf's Law can help to assess corpora by observing ranks of word frequency. According to Zipf's Law, in a standard corpus, the most frequent word in a corpus should occur two times more than the second frequent word in the corpus and so on. If a histogram is sorted by word's rank, with the most frequent words first, the shape of the curve is a Zipf curve (Manning & Schuetze, 1999: 24-30). Zipf's law says, if the Zipf curve plotted on a log-log scale, it must be a straight line with a slope of -1. Mandelbrot asserted that the Zipf curve is often a bad fit, abnormally for low and high ranks (Manning & Schuetze, 1999).

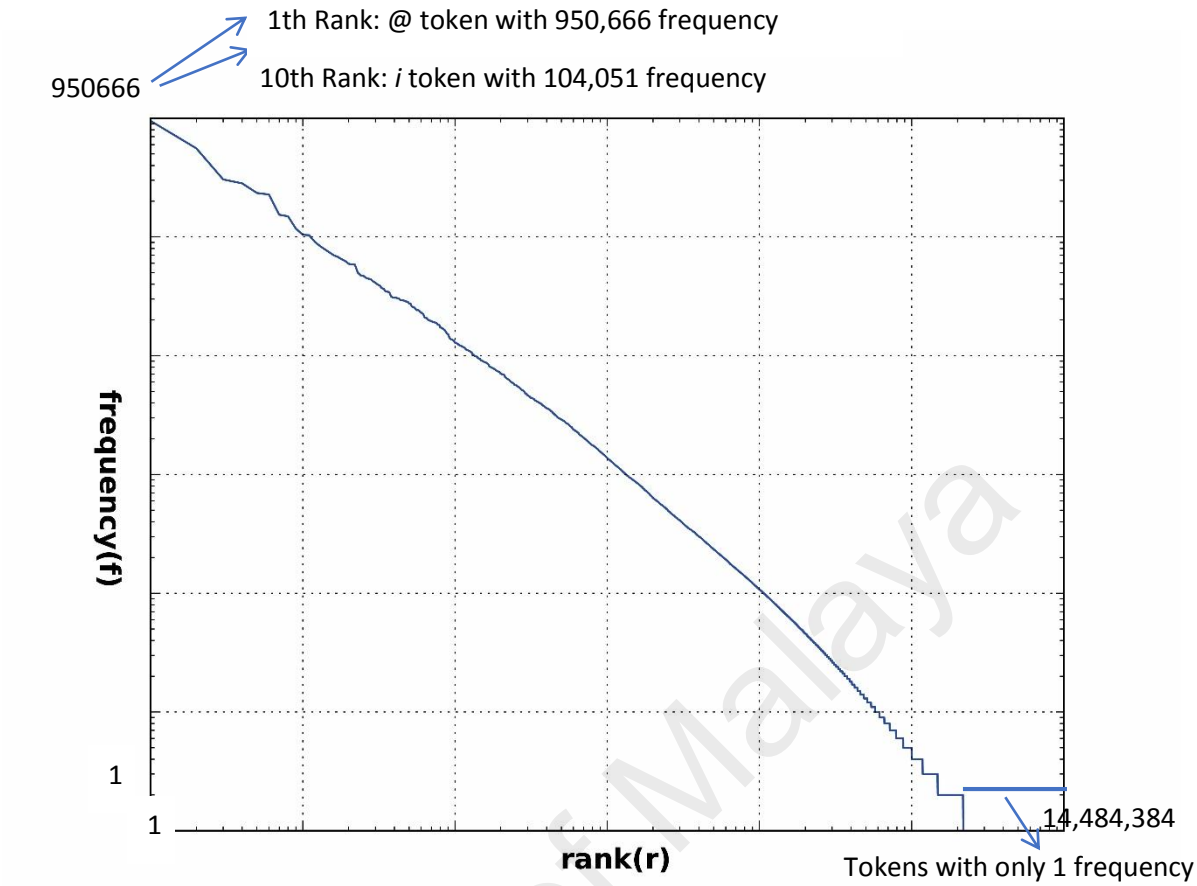

Figure 4.9: Zipf Curve for MCC Unigrams

Figure 4.9 shows that our corpus result is horizontal at the end of the graph. Therefore, approximately 1/8 of words appeared only once in MCC. APPENDIX F displays tokens with only one appearance in the corpus, which started with the letter m. 96% of one appearance words are OOV words, and does not exist in standard dictionary such as URIs, proper nouns, and misspelled words with repeated characters. The existence of the large number of misspelled words with repeated characters is common in chat-style text (Mapa et al., 2013). For example, the word "*mesti*" (i.e. *must* in English) can be written in infinite ways, such as *mestii, mesti, mestii, mestiii, mestiiii*, and so on.

4.3 Corpus Analysis

Our data analysis is based on TF-IDF schema (refer to Section 2.2.2.1). Before designing the normalization system, analyzing tasks was performed. All analyses are

performed on the Twitter corpus (i.e. MCC), except one analysis that is performed upon a standard Malay language corpus (i.e. *Dewan Bahasa dan Pustaka* corpus). *Dewan Bahasa dan Pustaka* (DBP) corpus is developed by *Dewan Bahasa dan Pustaka* which is a governmental institution in charge of monitoring and planning Malay language and literature in Malaysia. In 2008, the DBP corpus database comprises 128 million words which are compiled in 10 sub-corpora representing different genres of texts as follows: newspapers (e.g. Daily, Sunday editions, tabloids), books (e.g. Textbooks, academic books, novels, general reading), translations (e.g. Academic books and general readings), magazines (e.g. General and covers various fields), ephemerals (flyers, advertisements, brochures), drama, traditional texts, and poems. However, the development and enhancement of the corpus is ongoing (Rahim, 2014).

The initial analysis was to ascertain the frequency of distribution of unknown words in Malay language Tweets via TF-IDF. Therefore, the most prevailed unknown words and phrases are recognizable to formulate appropriate solutions for normalizing them. The result of the analysis stated that English words are the most frequent unknown words, and the abbreviations are the second most frequent. Moreover, more statistical analyses are conducted in order to know more detail about the abbreviation patterns. The last analysis was accomplished on DBP corpus in order to detect the patterns of consecutive repetition of letters in Malay words.

4.3.1 Frequency of Unknown Words

MCC is analyzed to enable us to calculate the distribution frequency of unknown words. Table 4.2 shows the top 20 most frequent words in MCC. Correct and standard terms that can be found in dictionaries are called In Vocabulary (IV). In the top 20, there are only five In Vocabulary (IV) words: Aku, ke, ada, dia, and yang. As shown in Table 4.2, the number of English words is six. Surprisingly, there are eight abbreviated

words and only one interjection. APPENDIX A list out the most 200 frequent tokens. The salient appearance of abbreviated words and English words in the top most frequent words in the corpus signify the enormous number of OOV words in Malay language Tweets.

Table 4.2: Top 20 Most Frequent Words

Rank	Word	English meaning	Description
1	I		English word
2	Aku	I	
3	Tak	Not	Abbreviated form of "tidak"
4	Nak	Want	Abbreviated form of "hendak"
5	The		English word
6	To		English word
7	You		English word
8	Ni	This	Abbreviated form of "ini"
9	Dah	Already	Abbreviated form of "sudah"
10	a		English word
11	Yg	That	Abbreviated form of "yang"
12	Tu	The	Abbreviated form of "itu"
13	La	Is (postfix)	Abbreviated form of "lah"
14	Ke	To	
15	haha		Interjection
16	Ada	Have	
17	Dia	He	
18	And		English word
19	Yang	That	
20	Kau	You	Abbreviated form of "engkau"

From Table 4.2, five categories of word forms are defined: IV words, words with extra repeated letters, English words, OOV words with special characters, and other types of misspelled words. Bahasa Wordnet, BNC, and Regular Expression (RE) is used to calculate the precentage of each catagory (refer to Section 2.2.2.1 for RE ellaboration). The narrow categorization is chosen because of the difficulty in distinguishing between types of misspelled words. The number of IV words is calculated using Bahasa WordNet (Karimah, Aziz, Noah, & Hamzah, 2011). To discover the size of fraction of

English words, all words are searched in BNC corpus. Table 4.3 refers to the percentage of each category in the corpus. Ergo, *60.2%* of MCC is composed of OOV words, and a large fraction of them are English words.

Table 4.3: Frequency of Types of Words

No.	Category	Percent Frequency
1.	IV words	39.8
2.	English words (OOV)	25.45
3.	Words with Consecutive letter repetition (OOV)	4.2
4.	Words with special characters from Thin-group (OOV)	0.5
	Words with special characters from Thick-group (OOV)	8.1
5.	Other types of Misspelled words (OOV)	21.95

To have a more precise understanding of OOV frequencies, special characters are divided into two groups: Thin-group (i.e. punctuations) and Thick-group (i.e. Other special characters). As shown in Table 4.3, the Thin-group represents the Malay punctuation letter (8 types). The members of the Thin-group are standard punctuations in Malay language. The Thick-group consists of special characters that are printed on ordinary computer keyboards, but excluding characters in the Thin-group (24 types). Table 4.4 shows that the percentage of frequency of OOV words which contain one of the special characters from the Thin-group is only 0.5%, which is in contrast with the percentage of frequency of those OOV words which contain one of the special characters from the Thick-group that is about 8.1%. It can be known from above percentage frequencies that words which contain one of special character from the Thin-group have low probability to be detected as OOV word.

Table 4.4: Special Character Categories

Group Name	Members	Size
Thin-group	*“!:?”,()*	8
Thick-group	*/<\>``;-_[{]}*&^%$#@+~=|_*	24

4.3.2 Abbreviation Patterns

Category No. 5 in Table 4.3 refers to "*Other Types of Misspelled Words*", which includes words abbreviation and word composition. The most frequent words in the MCC (see Table 4.2) refer to the tendency of users to abbreviate words. By scrutinizing abbreviated words, eight major types of abbreviation were discovered in chat-style Malay as shown in Table 4.5. An abbreviation detected only when the number of noisy token is lesser than correct word. Two linguists from University of Malaya have converted 7500 Tweets from MCC to their corresponding normalized Tweets (refer to Section 4.2.4.1). This dictionary also is explained in Section 5.2.3. A regular expression script identified 3,091 abbreviations out of 33,878 OOVs. Then, the linguist manually skimmed all the abbreviations and only eight abbreviation styles have been recognized.

Table 4.5: Abbreviation Patterns

Abbreviation	Type of Abbreviation	Example
Abbreviation 1.	Replacing tidak with the letter x	*tidak boleh: xboleh* (cannot)
Abbreviation 2.	Altering reduplication	*hari-hari: hari2* (days)
Abbreviation 3.	Eliminating vowel letters	*bangun: bgn* (get up)
Abbreviation 4.	Eliminating the letter r	*pergi: pegi* (going)
Abbreviation 5.	Eliminating affix	*kekasih: kasih* (sweet-heart)
Abbreviation 6.	Eliminating initial letter	*itu: tu* (that)
Abbreviation 7.	Eliminating last letter	*boleh: bole* (can)
Abbreviation 8.	Combining words	*macam mana: camne* (how)

- Abbreviation 1: It refers to the negation rule. In Malay, the grammatical rule for producing a negative sentence is the insertion of the word *tidak* before the verb. In the abbreviated form *tidak has been replaced with x.* This abbreviation type is also reported by by Basri et al. (2012).

- Abbreviation 2: It refers to the reduplication in Malay language. Reduplication is a morphological method for producing new meanings that

exists in many languages. For example, *hari* means *day* and *hari-hari* means *days*, and in the abbreviated form, it becomes *hari2*. This abbreviation type is also reported by by Basri et al. (2012).

- Abbreviation 3: The most of vowel letters are omitted. For example, *bangun* is converted to *bgn* (means *get up*).
- Abbreviation 4: Some *r* characters are omitted. For example, *pergi* is converted to *pegi* (means *going*).
- Abbreviation 5: Some affixes are omitted. The grammar of Malay language primarily based on affixes and suffixes. For example, *ke* is one of Malay noun affixes that will be removed in the abbreviated form (e.g. *kekasih* is converted to *kasih* (means *sweetheart*).
- Abbreviation 6: The first character is deleted. For example, *itu* is converted to *tu* (means *that*).
- Abbreviation 7: The last character is deleted. For example, *boleh* is converted to *bole* (means *can*).
- Abbreviation 8: The most complex abbreviation form is the combination of words. For example, *macam mana* is converted to *camne* (means *how*).

4.3.3 Collocation Frequency

To calculate bigram collocation in the MCC, Normalized Pointwise Mutual Information (NPMI) is employed, since it is one of the most efficient and accurate association measures (Bouma, 2009 and Pecina, 2008). NMPI is a new version of Pointwise Mutual Information where the results are normalized between [-1, +1]. Where -1 refers to never occurring together, 0 to independence, and +1 to complete co-occurrence.

Table 4.6: Top 40 Most Frequent Bi-gram Collocations in MCC

Rank	Phrase	English meaning	Description
1	terima kasih	thank you	
2	happy birthday	happy birthday	English phrase
3	im at	I am at	English phrase
4	have a	have a	English phrase
5	hari raya	Holiday/festive season	
6	selamat hari	happy days	
7	selamat pagi	good morning	
8	good morning	good morning	English phrase
9	good luck	good luck	English phrase
10	orang yang	those people who	
11	hari ini	today	
12	jgn lupa	do not forget	Acronym form of "Jangan lupa"
13	tak boleh	cannot	Acronym form of "tidak boleh"
14	nasib baik	Fortunately/luckily	
15	nasi lemak	glutinous rice	
16	sewaktu dengannya	with it/similar, same time	
17	macam mana	How	
18	next week	next week	English phrase
19	maaf zahir	apology	
20	tak pernah	Never	Acronym form of "tidak pernah"
21	open house	open house	English phrase
22	orang lain	other person	
23	tak payah	no need	Acronym form of "tidak payah"
24	solat fardhu	obligatory prayers	
25	nak buat	to do	Acronym form of "hendak buat"
26	kat mana	where?	Acronym form of "dekat mana"
27	aku nak	I want	Acronym form of "aku hendak"
28	lebih baik	better	
29	aku tak	I do not	Acronym form of "aku tidak"
30	take care	take care	English phrase
31	jangan lupa	do not forget	English phrase
32	may Allah	may God	English phrase
33	buat apa	for what	
34	Allah bless	God bless	English phrase
35	dah lama	It had been a long time	Acronym form of "sudah lama"
36	kirim salam	send regards	
37	masuk waktu	In/is time	
38	petaling jaya	Petaling Jaya (place name)	Proper noun

Table 4.6: Continued

Rank	Phrase	English meaning	Description
39	pakai baju	wear clothes	
40	waktu solat	prayer time	

Table 4.6 presents the top 40 most frequent bigram collocation in MCC. Amongst the top 40 most collocations, 21 of are OOV which comprises 11 foreign phrases, 9 abbreviated phrases and 1 proper noun. This shows that foreign phrases and abbreviated phrases are the most common collocations found in Malay language chat-style text. APPENDICES B and C show the top 200 most frequent bigram and trigram collocations, respectively.

4.3.4 Metadata Analysis

There are about 530 unique Twitter client applications in MCC, but the most popular Twitter mobile-based client applications are developed by Blackberry, Twitter, and Ubersocial. Approximately, about *23%* of Tweets are posted from a Twitter website, and the remaining *77%* are posted on other websites and applications. In summary, MCC users can inspect texts that have been sent from a particular Twitter application.

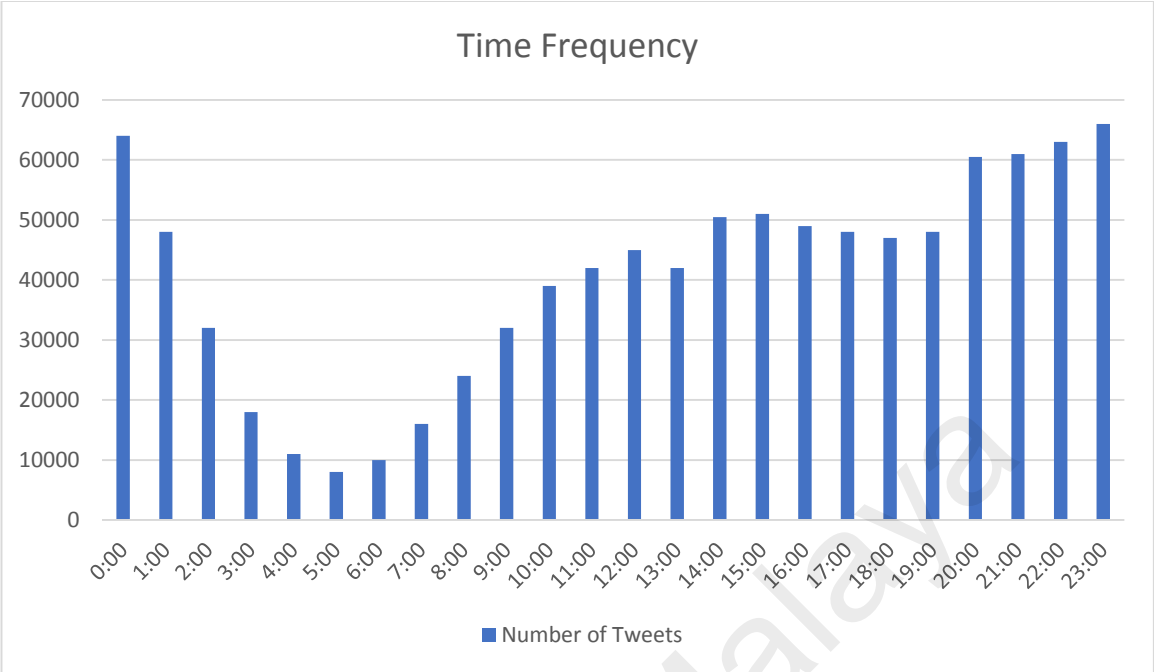

Figure 4.10: Time Pattern in MCC

Figure 4.10 depicts the number of Tweets that are sent around-the-clock. It is found that 23:00hrs is a peak time for sending Tweets. However, posting Tweets decreased dramatically from 23:00hrs to 06:00hrs, then it increased again from 07:00hrs to15:00hrs. It then fluctuated over the next five hours. It is observed that posting Tweets rose significantly from 15:00hrs to 23:00hrs. In conclusion, Twitter usage time patterns revealed peak time usage of Twitter by Malaysians of being between 20:00hrs and 24:00hrs. Usually 20:00hrs to 24:00hrs is for entertainment or relaxing. Users often send Tweets as things arise in real time, for example, while thinking about something, reading text, or watching television (Zhao & Rosson, 2009). Therefore, it is reasonable to conclude that a big portion of the MCC is constructed of texts, which represent light subjects and have the potential to include chat-style words.

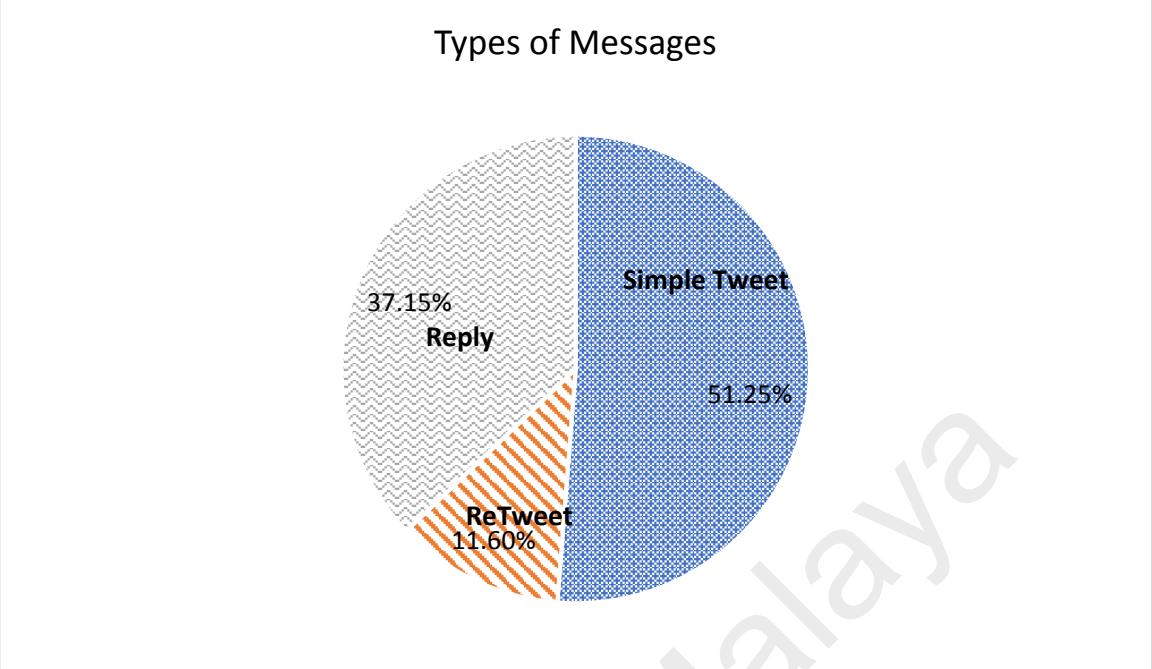

Figure 4.11: Percentage of Message's Type in MCC

As mentioned before in the compiling process in Section 4.2.1, only production texts were targeted. For this reason, users, who use Twitter for individual purpose, but not for broadcasting news or commercial messages, are selected. However, Figure 4.11 shows that only *11.6%* of the messages are Retweets, thus, this portion of corpus inevitably might include reception text. As a result, users who would like to consider only production text must ignore this portion of the corpus. The results show that about half of the messages are simple Tweet and more than approximately *37%* of Tweeter messages are Reply. A considerable number of Reply is a strong contender in proving that Malaysians use Twitter for chatting and conversation.

4.3.5 Miscellaneous Information

The corpus-driven analysis reveals variety of information about Malay language Tweets like distribution frequency of at-sign, number-sign, asterisk, and hyperlink but only period and letter case (capitalization) are described here for the sake of brevity. Moreover, around *68%* of Tweets contain periods, that is, 683,827 periods in

62,228,658 letters. In other words, about *4.72 %* of words contain periods. Number of periods as shown in Table 4.7, proves tendency of users to drop periods at the end of sentences. The result of majuscule letter analysis shows that about *49.4%* of Tweets begin with a majuscule letter. Table 4.8 shows the exact figures regarding letter cases in MCC.

Table 4.7: Periods in MCC

Type of letter	Quantity of letter	Percentage of letter
Total periods	683,827	1.09%
Period in last character of a word	300,832	0.48%
Period before EOL	262,151	0.42%

Table 4.8: Majuscule letters in MCC

Type of letter	Quantity of letter	Percentage of letter
Total majuscule letters	870674	1.39%
majuscule letter in begin of Tweet	494098	0.79%
Others	376576	0.60%

4.3.6 Letter Repetition in DBP Corpus

Category No. 3 in Table 4.3 refers to "*Words with Consecutive letter repetition*". One of the significant contributions of this work is that it presents a method for eliminating extra repeated letters from Malay words. To find morphological features of Malay word, an analysis is carried out over DBP corpus consisting of 135 million words (Jones & Ghani, 2000). DBP corpus comes with an online concordance web application to smooth frequency analyses such as TF-IDF. The TF-IDF formula (refer to Section 4.3 and Section 2.2.2.1) is applied at letter level not in token level. The result of the analysis shows that repetition of the same letter does not occur in Malay correct words except for nine certain conditions as shown in Table 4.9.

Table 4.9: Repeating Letters Only Appear in Specific Circumstances

No.	Description	Example
1.	borrowed words may include repeated letters (especially borrowed words from Arabic language).	Jemaah 'congregation'
2.	aa: a morpheme ended with 'a'+ '-an' (suffixes).	permintaan 'demand'
3.	ee: 'Ke-' (affix) + morpheme started with 'e'.	keempat 'fourth'
4.	ii: 'di-' (affix) + morpheme started with 'i'.	diisytiharkan 'declare'
5.	kk: a morpheme ended with 'k'+ '-kan' (suffixes).	meletakkan 'putting'
6.	kk: a morpheme ended with 'k'+ '-ku' (suffixes).	anakku 'my son'
7.	kk: a morpheme ended with 'k' + '-kah' (suffix).	masakkah 'is it ripe?'
8.	nn: a morpheme ended with 'n'+ '-nya' (suffixes).	kemudiannya 'then'
9.	ll: a morpheme ended with 'l' + '-lah' (suffix).	betullah 'that is correct'

The format of Malay morpheme is Consonant-Vowel-Consonant, where Consonant is optional (Kadir et al., 2011). In contrary to English, the syntax of Malay language is simple in terms of repeating letters in a row. Few borrowed Arabic words have two of the same letters in sequence. 452 borrowed words are gathered and kept as a bag of words. APPENDIX E is displaying most of them along with their English translation and origin language. While the other eight conditions concern word affixation. Therefore, only *aa, ee, ii, kk, nn,* and *ll* exist in Malay words and strictly in specific circumstances. The *aa* appears when a morpheme that ended with *a* is added to *an* suffix. The *ee* appears when a *ke* affix is added to a morpheme that started with *e*. The *ii* appears when a *di* affix is added to a morpheme started with *i*. The *kk* appears when a morpheme ended with *k* is added to a *kan* or *ku* or *kah* suffix. The *nn* appears when a morpheme ended with *n* is added to a *nya* suffix. Finally, the *ll* appears when a morpheme ended with *l* is added to a *lah* suffix.

4.4 Summary

This chapter described the process of constructing a Twitter corpus known as the Malay Chat-style Corpus (MCC), which is a representation of chat-style Malay text. In

other words, the target population of the corpus is Malay chat-style text. A sampling frame that consists of 4,500 public Twitter user IDs is constructed using Twitter APIs. The user IDs in the sampling frame belongs to users who chose Malaysia for their profiles. According to the purposive sampling technique, a linguistic expert investigates the sampling frame of 4500 Malaysia Twitter users to detect 321 users who use informal Malay language in their messages. Finally, 3,200 messages are fetched from each user via Twitter APIs. Therefore, in the process of compiling the corpus, variety is assured through selecting 321 authors, which signifies considering 321 different writing styles. We drew the Zipf curve for MCC to prove:

1. The most frequent word in a corpus occurred two times more than the second frequent word in the corpus and so on, that is the one of the main characteristics of a robust corpus.

2. Considerable number of words in the corpus (i.e. 12.5% of the words) appears only once. Further investigation proves that 96% of them are OOV.

The MCC is a synchronic corpus, because the gathered messages belong to a particular time, which is assigned to a date between January 2012 and November 2012. To protect the privacy of the Twitter users, the anonymizing process was accomplished by applying the MD5 hash algorithm on the user IDs. Moreover, our corpus analysis reveals the characteristics of the MCC. The most frequent words and the most frequent collocations in the corpus include a significant number of OOV words, in particular, foreign and abbreviated phrases. The metadata analysis reveals three aspects of the corpus, namely: 1) around *23%* of Tweets were posted from the Twitter website, 2) the peak time of Twitter usage by Malaysians is between 20:00hrs and 24:00hrs, and 3) about half of the messages are simple Tweets, more than *37%* of them are replies, and around *11%* are Retweets.

Two different methods are employed to evaluate the representativeness of the corpus: cartography and automatic language identification. The cartography evaluation proves that messages are sent from Malaysian soil. Langid is an open source Python library for detecting variety of languages by employing a probabilistic language identification methods (Lui & Baldwin, 2012). Via Langid, it has been proven that the language used in the messages is Malay. Future work should focus on improving the quality of the corpus, e.g., by annotating the corpus (either automatically or manually) with part-of-speech tagging or by adding translations into other languages.

The next chapter discusses our normalization architecture based on the achieved analyses results. The frequency of unknown words shows that which types of noisy words should be considered. From the most common unknown words, it is understood that we should develop dedicated modules for English words and repeated letters in the architecture. Without the letter repetition analysis, eliminating extra letters would be impossible. In addition, abbreviation pattern analysis helps to clarify addressing abbreviation styles such as token's duplication (e.g. Converting *buku2* to *buku-buku* '*books*') and token negation (e.g. Converting *xsenang* to *tidak senang* '*not happy'*). Finally, our colloquial dictionary is built based on the results of the collocation analyses.

CHAPTER 5: AN ARCHITECTURE FOR MALAY LANGUAGE TWEET NORMALIZATION

5.1 Introduction

The design of the proposed normalization architecture is based on the results of corpus analysis. Chapter 4 discusses the process of building corpus and the results of the corpus analysis. This chapter elaborates the proposed architecture, which encompasses seven modules: 1) enhanced tokenization, 2) IV words detection, 3) colloquial dictionary lookup, 4) repeated letters elimination, 5) abbreviation normalizer, 6) English word translation, and 7) De-tokenization. This chapter describes how each module is formulated based on the results of the corpus analysis. For example:

- Section 4.3.6 elaborates the analysis upon DBP corpus shows that there is no repeated letters in standard Malay language except in eight prefix/suffix related conditions (refer to Table 4.9). Therefore, a repeated letters elimination module is formulated based on the pattern of repeated letters in Malay language.

5.2 Architecture of Malay Language Tweet Normalization

The proposed architecture comprises of seven modules. Each module was implemented via a Finite-State Machine (FSM). A finite-state machine, or FSM for short, is a model of computation based on an imaginary machine of one or more states. Only one single state can be activaed at the same time. To execute different tasks, the machine must travel from one state to another. Figure 5.1 displays a simple FSM for door closing/opening tasks. In our architecture, we have 13 states or tags for each token as shown Table 5.1. The proposed architecture is based on a deterministic FSM, where, every state has exactly one transition for each possible input. Each normalization module is a transition that may cause state transitions for tokens. Figure 5.2 refers to the orders of modules in the architecture.

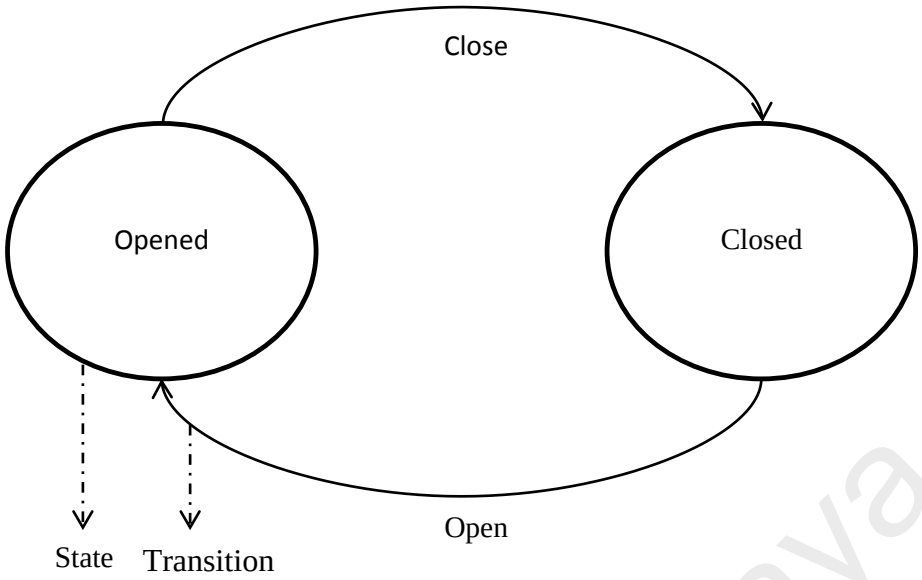

Figure 5.1: Example of Door Finite State Machine

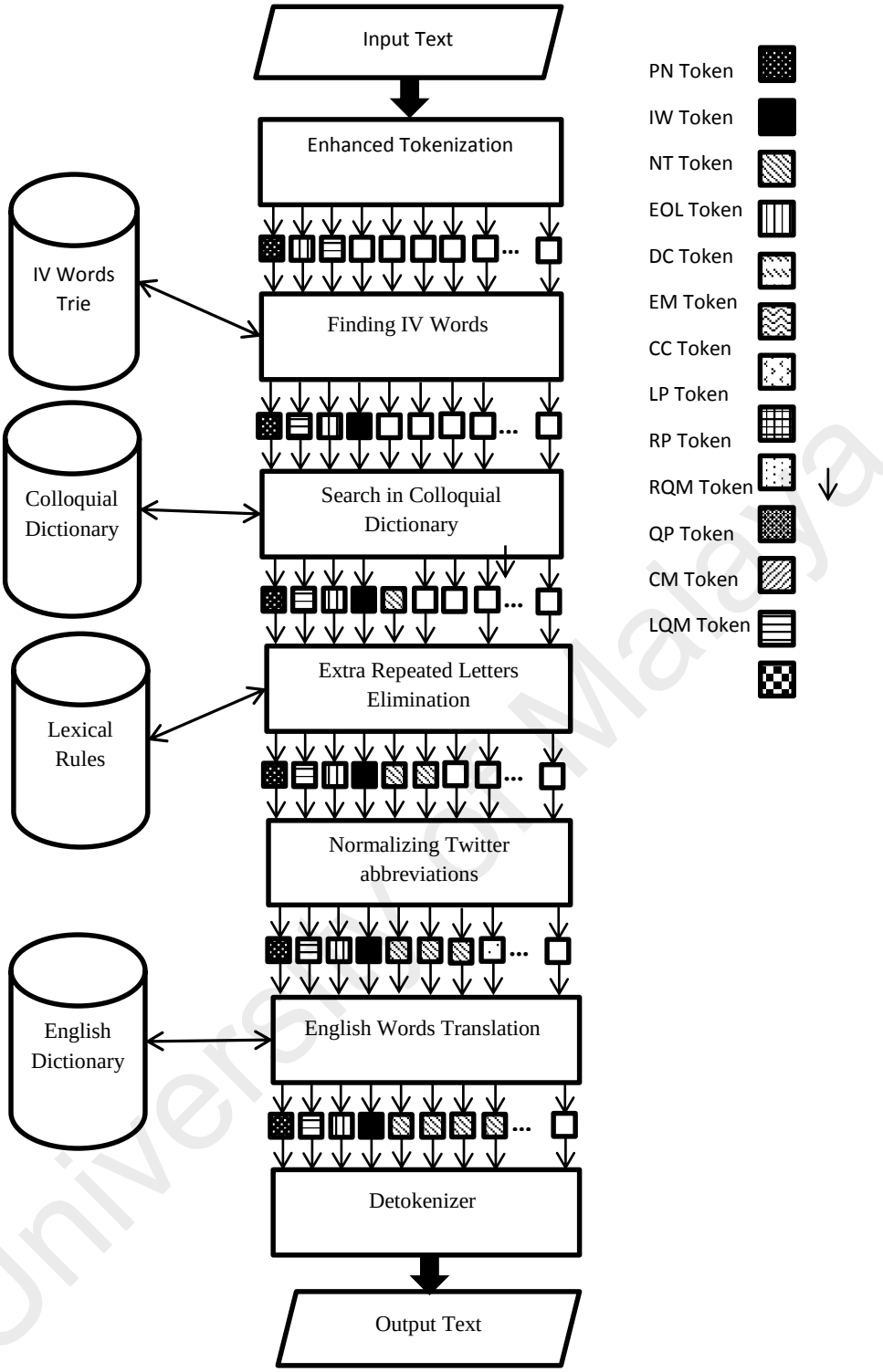

Figure 5.2: Normalization Architecture

There are 13 types of states that are used in the architecture as shown in Table 5.1. Each module assigns different sets of tag/state to the tokens. There are three main tags/states that are defined, which are PN, IW, and NT. These tags/states are referring

to proper nouns, In-vocabulary words, and normalized tokens respectively. There are ten other tags that deal with special characters such as End of Line (EOL) and Dot Character (DC), as listed in Table 5.1. Details of each module are discussed in the next subsections.

Table 5.1: Types of States

No.	State/Tag	Full Name	Description
1.	PN	Proper Noun	Assigned to proper names protect from further changes.
2.	IW	In-vocabulary Word	Assigned to In-vocabulary tokens protect from further changes.
3.	NT	Normalized Token	Assigned to normalized tokens protect from further changes.
4.	EOL	End of Line	Assigned to last token of a line.
5.	DC	Dot Character	Assigned to tokens with period at end.
6.	EM	Exclamation Mark	Assigned to tokens with exclamation mark at end.
7.	CC	Colon Character	Assigned to tokens with colon at end.
8.	RP	Right Parenthesis	Assigned to tokens with close parenthesis at end.
9.	LP	Left Parenthesis	Assigned to tokens with open parenthesis at begin.
10.	LQM	Left Quotation Mark	Assigned to tokens with open quotation mark at begin.
11	RQM	Right Quotation Mark	Assigned to tokens with close quotation mark at end.
12.	QP	Question Point	Assigned to tokens with question point at end.
13.	CM	Comma Mark	Assigned to tokens with comma at end.

5.2.1 Enhanced Tokenizing

Similar to most NLP projects, the first module in the normalization architecture is tokenization. An enhanced tokenization module is designed based on the result of our Malay language Tweets analysis, where, words that contain one of the special characters from Thin-group (see Table 4.3) has low chance to be considered as a proper noun. Apart from breaking Tweets up into tokens, the tokenization module detects particular types of proper nouns and tags them with PN tag.

Previous researches did not consider specialized tokenizer for Tweets, though tokenization is one of the most important parts in text preprocessing (Webster & Kit, 1992). Conventional tokenization algorithms (e.g. breaking it up at white-space and punctuation marks) cannot be used for the noisy text due to its erratic characteristics. Thus, an enhanced Tweet tokenization algorithm is introduced. The tokenization module helps to discriminate between OOV words and ill-formed words through detecting the proper nouns which contain special characters. The tokenization algorithm removes punctuations and converts spaces to new lines. The de-tokenization module undoes the effects of this module. In Basri et al. (2012), punctuation elimination has been done after tokenization which cause reshaping the original inputs and generating incomplete outputs.

The enhanced Tweet tokenization algorithm shown in Figure 5.3 and Figure 5.4 is designed based on characteristics of Malay language Twitter messages. Tweets are prone to have an enormous number of proper nouns such as usernames and hyperlinks. Proper nouns are detectable by locating special characters and digits in tokens. As shown in Table 5.1, there are nine tags that are designed for different symbols known as exclamation marks (!), colon signs (:), question marks (?), left quotation marks (“), right quotation marks (”), comma marks (,), period characters (.), left parenthesis ((), and right parenthesis ()), as introduced in Section 4.3.1 as a Thin-group. According to the OOV word analysis; these symbols are very common in standard writing style, while other symbols occur in proper nouns. For example, at-sign (@) and number-sign (#) refer to Twitter username and Twitter topic respectively. The initial letter of sentences and names may not be in majuscule letter, and majuscule letters might be used to signify emphasize and emotion such as ‘*dia berkata BOLEH’*. Therefore, identifying the end of a sentence is an arduous task, although End-of-Line (EOL) can be detected by finding the EOL special characters.

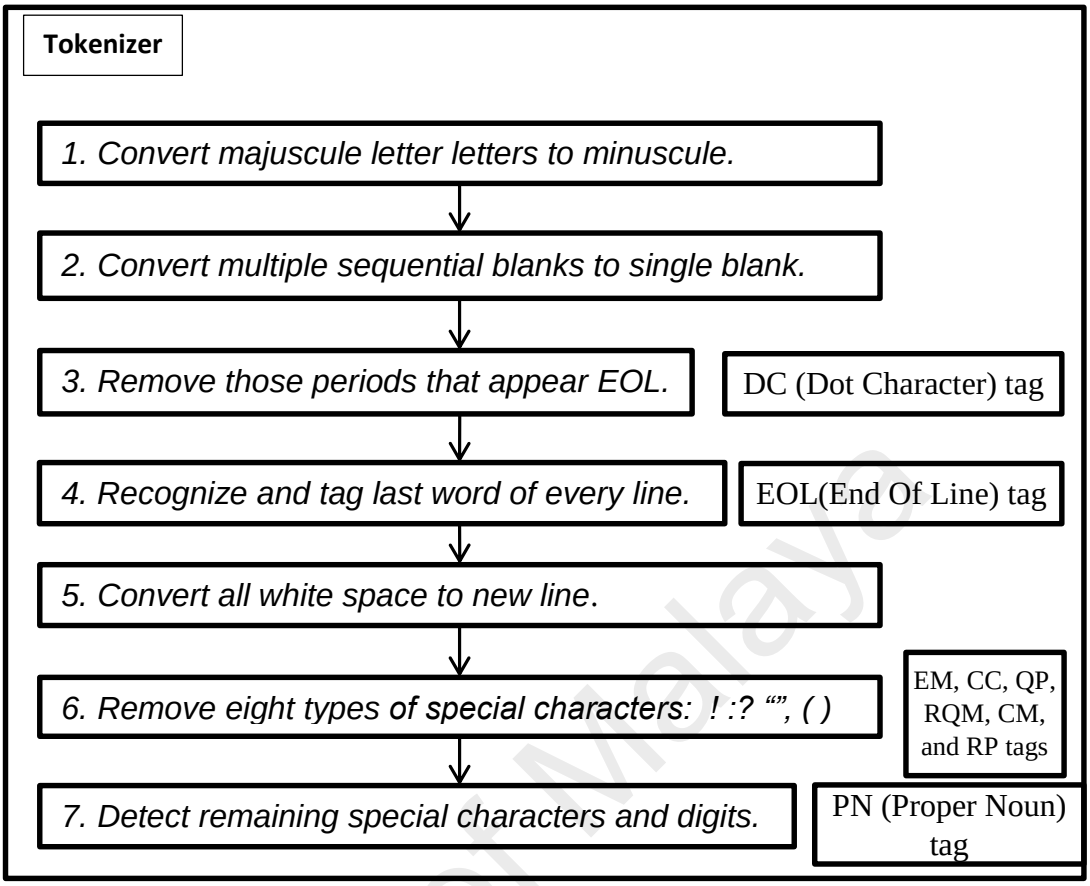

Figure 5.3: Tokenization Module

```
start
WHILE uppercase letters exist DO
    convert uppercase letters to lowercase
WHILE multiple sequential blanks exist DO
    convert multiple sequential blanks to single blank
WHILE periods appear End Of Line DO
    remove those periods that appear End Of Line
    add DC tag
WHILE last word of line is not labeled DO
    recognize and tag last word of every line
    add EL tag
WHILE any white space exists DO
    convert all white space to new line
WHILE eight types of special characters:  ! :? “”, ( ) exist DO
    remove ! :? “”, ( )
    add EM, CC, QP, RQM, CM, and RP tags
WHILE any special characters and digits exist DO
    detect remaining special characters and digits
    add PN tag
end
```

Figure 5.4: Tokenization Pseudo Code

First, all majuscule letters are converted to minuscule because majuscule letter does not have any orthographic value in noisy texts. After converting consecutive extra blanks to single blank, period marks will be eliminated in the third step. The results of the analysis prove that most of the sentences do not end with a period. Therefore, in our system, if a period does not appear before EOL characters, it will appear in a proper noun. In other words, if the last word of the line contains a period at the end, the period will be eliminated and the word will be labeled with DC (Dot Character) tag. The fourth step is to delete all EOL characters and add the EOL tag to the last word of lines.

After converting white spaces to new lines, each line will be considered as a token. In the sixth step, if the last character of a token is the exclamation mark, colon sign, question mark, right quotation marks, comma mark, or right parenthesis, the character will be deleted and appropriate tags (EM, CC, QP, RQM, CM, or RP) will be added to it. The de-tokenization module uses the tags to append the symbols to the tokens. In addition, if the first character of a token is left quotation marks or parenthesis, the character is removed and LQM or LP tags are added to it. The last step is to detect proper nouns. If any character, except for alphabetic characters and digit *2* ― 2 has special usage in Malay lingo― appears in a token, the token will be tagged with PN tag.

5.2.2 Finding In-Vocabulary Words

After tokenization, detection of IV words is the next, and tagging them with an IW tag to protect them from further alteration. Before the normalization, IV and OOV words need to be distinguished. This module receives tokens and then later searches them in a Trie data structure. To achieve a fast search, all IV words are inserted from Bahasa Wordnet (Noor et al., 2011) text file to q-fast Trie data structure. Q-fast needs $O(N)$ space and $O(\sqrt{\log M})$ time of retrieval, insertion, and deletion (Willard, 1984). To have

factual and expeditious searches, only tokens that do not have the PN tags are searched. If the token is found in the q-trie, IW tag will be attached.

Trie or prefix tree enables high speed longest-prefix matching. Since a limited number of IV words are available in digital form, maximum-prefix-length match is used to boost the recall and enhance the coverage. Therefore, q-fast is traversed using characters of the input token. If a token's prefix matches a word, tionarythe module stores the current length, and looks for a longer match, then, the longest match will be returned. To achieve a high precision score, a threshold of the prefix length is defined. Different threshold values will be tested in the evaluation stage. As an example, suppose Figure 5.5 is the data structure, where threshold value is set to 3, and input token is *toes*. While there are no *toes* in our data, the module distinguishes the token as IV word, and adds an IW tag to it.

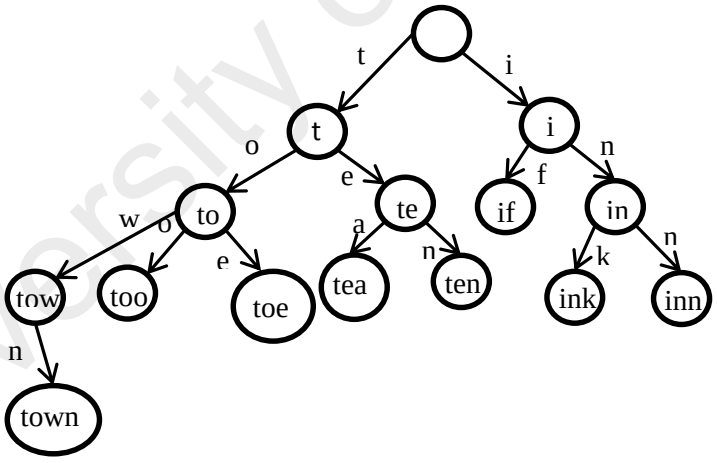

Figure 5.5: Q-trie Data Structure

5.2.3 Search in Colloquial Dictionary

Using a special dictionary, the third module coverts the most common noisy phrase (e.g. *yg —> yang, ni —>ini, tak —>tidak,* and *nak —>hendak*) into standard form and attaches NT (Normalized Token) state to the converted words. Previous studies have shown that a dictionary based systems can outclass complex approaches (Aw & Lee,

2012; Clark & Araki, 2011; Han et al., 2012). This chapter also describes the process of compiling the dictionary. The noisy term dictionary has been utilized in a previous work to address normalization of Malay text (Samsudin et al., 2012). However, using a dictionary cause ambiguity in a situation where the word can have more than one equivalent in standard English. For Example, word '*2*' can be converted to '*itu*', or '*dua*' according to their context. Thus, in this research; the work was modified to overcome the ambiguity problem by using a context-aware method.

To build a context-aware colloquia dictionary, OOV words are collected from MCC along with their following and preceding tokens. To build the dictionary, two linguists have converted 7500 Tweets from MCC to their corresponding normalized Tweets (refer to Section 4.2.4.1). The compiled dictionary encompasses 33,878 OOV instances. Then, the occurrences of each word group (preceding word, ill-formed word, following word) are counted, and, word groups that appeared more frequently (i.e. two or more than two times), were inserted into an XML structure.

Because of the small number of entries and simple scheme of the dictionary, Python dictionary data structure (Martelli, 2003) is employed to implement the dictionary. Python Dictionary is mutable, consists of pairs (called items) of keys and their corresponding values, where the keys are unique in a dictionary. OOV words and their surrounding words were inserted into the keys, and their normalization equivalents into values. Figure 5.6 refers to an example of Python dictionary items.

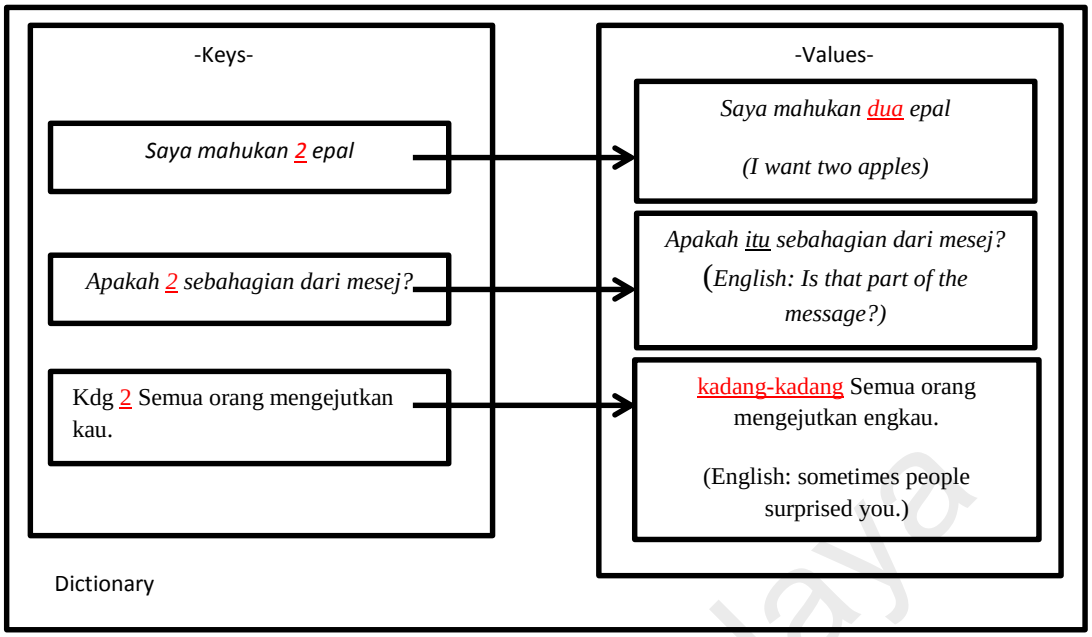

Figure 5.6: Dictionary Data Structure

After selecting high frequent phrases, the dictionary encompasses more than 765 entries. Table 5.2 displays sample entries of the dictionary where '*nk*' can be converted to '*anak*' or '*hendak*' according to their context. If a token does not have PN (Proper Noun) and IW (In-vocabulary Word) tag, it will be merged with its preceding and following tokens; and the phrase is searched in the dictionary. For example, "*keluar/IW pn tidak/IW*" will be converted to "*keluar pn tidak*", and searched in the dictionary. If a phrase is found, the middle token will be replaced with its meaning, and Normalized Token (NT) tag will be assigned to it to prevent further normalization process in the next modules.

Table 5.2: Colloquial Malay Dictionary Example

No.	Key: Noisy Phrase	Value: Standard Phrase
1.	xdela sgt	tidak adalah sangat
2.	sng sgt	senang sangat
3.	Banyak sgt akaun	Banyak sangat akaun
4.	Kenal sgt dah	Kenal sangat sudah
5.	sy sgt tak	saya sangat tidak
6.	Sgt beruntung org	sangat beruntung orang

Table 5.2: Continued

No.	Key: Noisy Phrase	Value: Standard Phrase
7.	wanita2 sgt kan	wanita-wanita sangatkan
8.	jgn kuat sgt	jangan kuat sangat
9.	Sgt cantekk n	sangat cantik dan
10.	comell sgt laa	comel sangatlah
11.	syukur sgt2..	syukur sangat-sangat..
12.	cz nk tgk	sebab hendak tengok
13.	Mana nk cari	mana hendak cari
14.	kak nk tnya	kakak hendak tanya
15.	I nk cpt	saya hendak cepat
16.	macam xnk	macam tidak hendak
17.	nk blnja on	hendak belanja boleh
18.	mnjadi nk soleha	menjadi anak soleha
19.	semoga nk sis	semoga anak kakak
20.	nk org la	anak oranglah

5.2.4 Eliminating Repeated Letters

In Twitter sphere, extra repeat letters are used to signify pronunciation, stress or other emotions such as anger such as '*Tidakkk Bolehhh'*. Thus, removing redundant letters is a crucial task for all Latin alphabet writing systems. One of the significant contributions of this work is that it presents a method for eliminating iterated letters from Malay words. The result of our analysis indicates that there are no any repeated letters in the Malay writing system except for nine specific conditions. The conditions have converted to word patterns. Section 4.3.6 elaborates the analysis upon DBP corpus shows that there is no repeated letters in standard Malay language except in eight prefix/suffix related conditions (refer to Table 4.9), and appearance of borrowed words. Therefore, a repeated letters elimination module is formulated based on the pattern of repeated letters in Malay language. The proposed architecture in the fourth module uses pattern finder to distinguish and remove extra letters.

A novel method to tackle the letter repetition in Malay words is proposed. The MCC analysis in Section 4.3.6 proves that the standard Malay language does not possess any repeated letter, except for a few identified conditions, which are related to affixes. The letter repetitions exceptional conditions (refer to Table 4.9) have converted to Regular

Expressions (RE). This module uses a search engine to check if a word matches the patterns. Figures 5.7 and 5.8 show steps of this module that are three:

I. The first step is to detect tokens with letter repetition in a row.

II. In the next step, borrowed words (i.e. loan words) will be detected and tagged as IW via a bag of borrowed words. Section 4.3.7 and APPENDIX E refer to borrowed words.

III. After searching in the affixes patterns, extra letters will be eliminated accordingly.

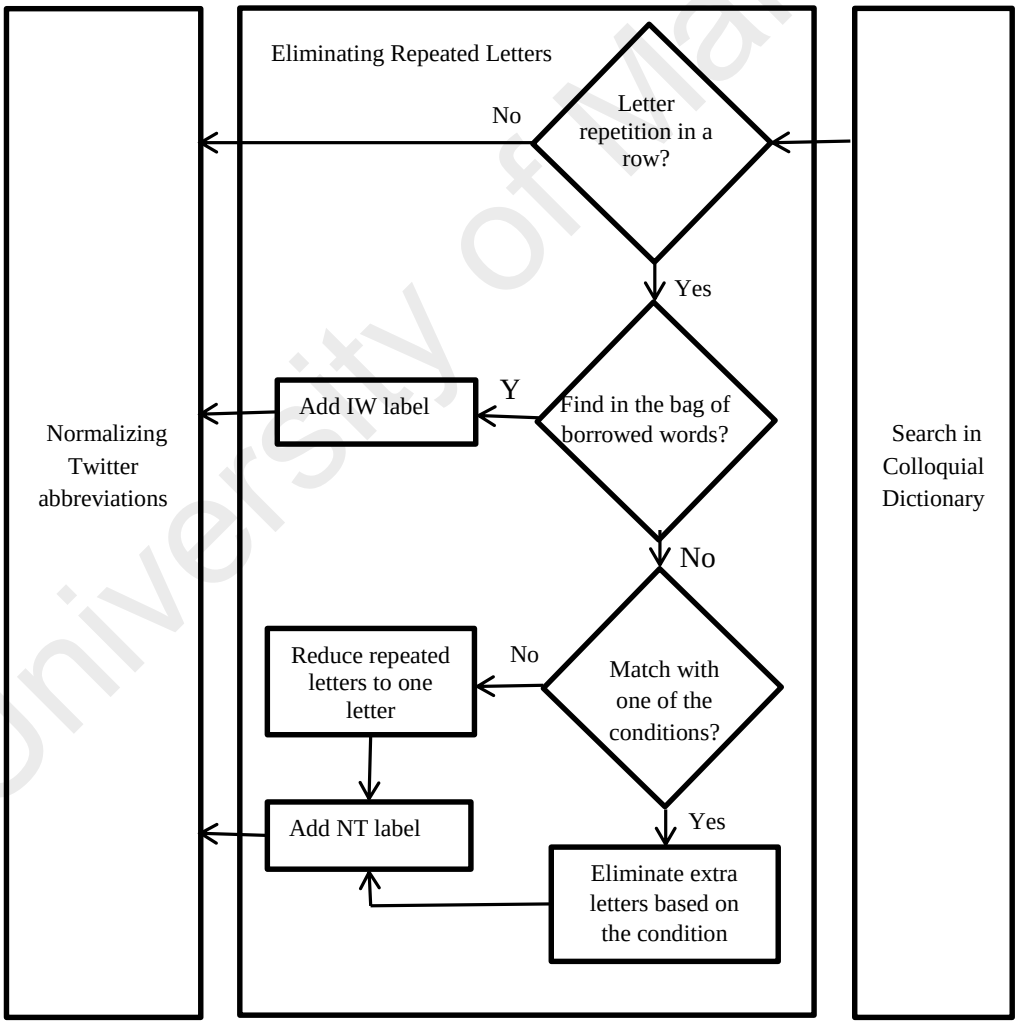

Figure 5.7: Repeated Letter Elimination Module

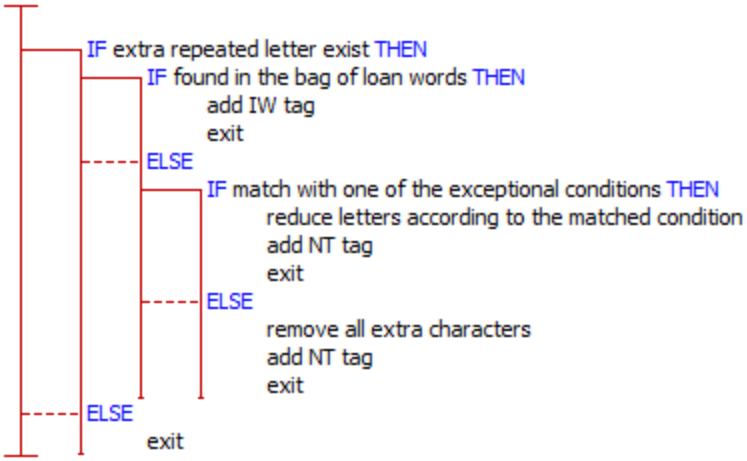

Figure 5.8: Repeated Letter Elimination Pseudo Code

In this module, RE is used in determining if tokens match to one of the conditions. Extra letters are eliminated based on the matched pattern. If a token match with one of the conditions, the repeated letters will be eliminated in a certain way that it does not disturb the pattern. For example, the token '*anakkkuuuu'* is transformed into '*anakku'* because it matched with the Condition 7 in Table 4.9: "*a morpheme ended with k attached to ku suffix*". By contrast, the token '*sayyaaaaaa'* is transformed into '*saya'* because it does not match with any pattern, thus, all repeated characters are reduced to only one.

5.2.5 Normalizing Abbreviations

The normalization architecture converts few Malay texting lingo writing styles to the correct forms. Our analysis shows two certain features of Malay blogging style, which are:

I. Abbreviating opposite words (e.g. *xboleh —> tidak boleh*)

II. Reduplications (e.g. *pura2 —> pura-pura*)

Thus, the fifth module detects and corrects the two blogger styles. Basri et al. (2012) describes a Malay normalization approach which availed predictable Malaysian blogger writing styles, that is, the regional writing style (Selangor lingo writing style)

is addressed. Although, the abbreviations of opposite word and reduplication have resolved by Basri et al. (2012), in this research the method is modified to achieve more accurate results using FSM. As explained in Section 5.2, in deterministic FSM, each token can have only one state that prevents unnecessary token modification. For example, '*Saya mahukan 2 epal*' will be converted to '*Saya mahukan dua epal*' but not '*Saya mahukan epal-epal'*.

This module addresses the Malaysian blogger abbreviation style. It normalizes tokens based on the analysis of the patterns of abbreviations in Section 4.3.2. It concentrates on two common abbreviation types in Malay lingo. As depicted in Figure 5.9, the input of this module is tokens, which do not have any of PN, NW, IW, and NT tags to ensure that the input are not digits and special names. Next, if the last character of a token is the digit *2*, the digit will be replaced with the token's duplication preceded by a hyphen as shown in the example below:

Token: *buku2* will be transformed into *buku-buku* (*books*))

In addition, if the first character of a token is letter *x*, the letter will be replaced with *tidak* followed by a blank. For example:

Token: *xsenang* will be transformed into *tidak senang* (*not happy*))

Finally, modified tokens are tagged with the NT tag.

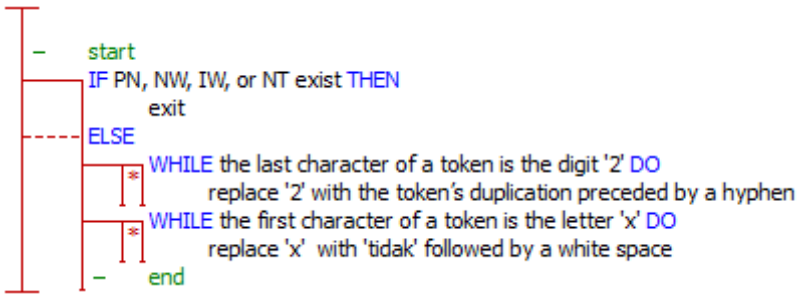

Figure 5.9: Abbreviation Normalizer Pseudo Code

5.2.6 Translating English Word

Code-switching between Malay and English is very frequent in Malay language Tweets. Bakar (2009) provided an evidence on the existence of code-switching in contemporary Malay language. Malay words are juxtaposed with English in the colloquial language called Manglish (Malaysian English). In addition, our analysis on one million Malay language Tweets show that about 32.9% of tokens are English words. The sixth module of the normalization architecture converts the English words into the Malay language. Basri et al. (2012) removes English words which cause the loss of sentence meaning and structure.

The sixth module of the architecture converts English words into Malay. Via Smith Malay-English Dictionary (Smith & Padi, 2006), this module transforms English words into Malay. The Smith Malay-English Dictionary text file is converted to a Python dictionary data structure (Martelli, 2003). Tokens, which do not have a PN (proper nouns), IW (In-vocabulary), and NT (normalized) tags, are detected. The detected tokens are looked up in the English-Malay dictionary and replaced with their meaning.

5.2.7 De-tokenizing

The last module removes the tags and performs de-tokenization on the normalized text. In the last step, de-tokenizing is an essential module that undoes the tokenization. As depicted in Figure 5.10, tokens, which have RQM or RP, are detected and quotation mark or open parenthesis will be added to their beginning. Secondly, the module detects tokens with EM, CC, QP, LQM, CM, and LP tags and adds appropriate special character to the end of tokens: exclamation, colon, close quotation, question, close parenthesis, and comma marks. Thirdly, all carriage returns will be transformed into

white spaces. Fourthly, a period character followed by a carriage return created after tokens with EOL tag. Lastly, all tags are removed.

```
start
WHILE OQM or OP tags exist DO
    add quotation mark or open parenthesis respectively to the begining of the tokens
WHILE EM, CC, QP, CQM, CM, or CP exist DO
    add appropriate special character to the end of tokens
WHILE carriage returns exist DO
    convert theem to white spaces
WHILE EL tags exist DO
    add carriage return preceded by a dot character to the end of tokens
WHILE any tag exist DO
    remove the tags
end
```

Figure 5.10: Detokenization Pseudo Code

5.3 Summary

This chapter discussed the proposed Malay language Tweet normalization architecture. The architecture is designed according to the MCC analysis in Chapter 4. The analysis shows that English words, abbreviation, and extra repeated letters have the highest frequency between OOV words. Thus, the architecture encompasses these modules:

i. Enhanced tokenization: a tokenization algorithm is formulated based on the features of Malay language Twitter messages. The algorithm can detect a few types of proper nouns bases on the type of special character in the token. The analysis in section 4.3.1 suggests that a few special characters appear more frequently in proper nouns than OOV words.

ii. IV word detection: instead of OOV detection, IV words will be detected by this module to protect them from further normalization. A dictionary of standard Malay words is utilized to detect the correct words.

iii. Colloquial dictionary: the most frequent collocations from the MCC corpus are inserted in a context-aware dictionary to covert colloquial and slang tokens.

iv. Repeated letter elimination: this module eliminates all extra repeated letters from tokens. The analysis in Section 4.3.6 illustrates that there are no repeated letters in standard Malay language, except for a few certain conditions that come from affixations.

v. Abbreviation normalization: this module normalizes two types of abbreviations in Malay language Twitter lingo: reduplication and negation.

vi. English word translation: an English-Malay dictionary is used to address the code-switching.

vii. De-tokenization: this module eliminates all tags and extra meta-data to convert back the appearance of the data to sentences and paragraphs.

CHAPTER 6: EVALUATION

6.1 Introduction

This chapter describes the evaluation methods that are used in this research. BLEU metric is exploited in this research to evaluate the proposed normalizer because the machine translation community trusts the BLEU evaluation metric, and it is the most established method (Callison-Burch, Osborne, & Koehn, 2006; Koehn & Monz, 2006). The chapter also proposed a normalizer using MCC corpus. To be able to compare the proposed normalizer with other probabilistic normalizers against the same dataset, a Statistical Machine Translation (SMT) normalizer is implemented and evaluated using MCC corpus. SMT addresses context sensitive text by considering noisy text as the source language and standard text as the target language. Finally, this chapter also provides the comparison of the SMT and the proposed architure, which proves the proposed normalizer can outperform others at least for Malay language normalization.

6.2 Evaluation Methods

In pattern recognition and information retrieval using binary classification, precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. Both precision and recall are based on relevance. For example, let's say there is an image processing computer program for capturing and identifying a number of cats in yards. For example, there is a yard that contains 9 cats and a few chicks and the program identified 7 cats. It is found that 4 of the identifications are correct, and 3 chicks are wrongly detected as cats, thus, the program's precision is *4/7* while its recall is *4/9*.

Word Error Rate (WER) is suitable when the length of source and target text is the same (Klakow & Peters, 2002). Usually, precision and recall are considered together to address length-related issues by applying n-gram precision. However, in BLEU metric,

multiple reference translations with different word choice can be utilized together. In addition, a good result only matches one of possible references, and not all. However, matching with all references signifies a bad translation as shown in Example 6.1.

Example 6.1:

A Malay sentence, '*Saya boleh belajar esok' can be translated into three references of Englsih sentences:*

Reference 1: *I may study tomorrow.*

Reference 2: *I can study tomorrow.*

Reference 3: *I can learn tomorrow.*

Example of machine translations can be:

Candidate 1: *I can may learn study tomorrow.*

Candidate 2: *I can study tomorrow.*

The tokens in the Candidate 1 matches with all the references and produce the highest recall. The tokens in Candidate 2 only matches with the Reference 3, therefore, produces lower recall than Candidate 1. However, Candidate 2 is a better translation than Candidate 1. Thus, one could match with all the gold standards and produce a high recall and poor translation. Therefore, a complicated formula is required to address multi-reference evaluation.

Utilizing BLEU score in normalization researches is a well established practice (Aw et al., 2006 and Oliva et al., 2013). Usually, the goodness of translations is judged by utilizing the BLEU score. The BLEU score is a tool designed for evaluating the accuracy of translations from one language to another. A BLEU score requires a gold standard, which contains the translations done by human. The gold standard is compared against a machine-translated version and is then assigned a score between zero and one. A score of one would indicate that the machine-translated version is the same as the human translated version, while zero means that the two versions are very

different. The language of a Tweet is so different from the normalized result that this tool should provide an accurate indication of how well the translation worked.

6.2.1 BLEU: Bilingual Evaluation Understudy

The BLEU score is from zero to one, where, zero signify no similarity between reference and candidate, and 1 signifies identical reference and the candidate. Therefore, the best candidates may not obtain score one. If there is more than one reference in the gold standard set, usually candidates obtain a higher score (Papineni, Roukos, Ward, & Zhu, 2002).

Usually, there are more than one correct translation references for a sentence. The reference translations are different in terms of word choice and word order. Although it is easy for human to select the better translation between references, it is a complex problem in MT (Papineni et al., 2002). However, this problem does not exist in the normalization field because there is only one correct translation of a noisy text. In Example 6.2, two candidate translations of a Malay source sentence (i.e. '*Parlimen adalah badan tertinggi negara ini*.') are depicted:

Example 6.2:

Candidate 1: *Parliament is the highest body of a nation.*

Candidate 2: *Assembly is the largest part of the country.*

Reference 1: *Assembly is the highest body of the nation.*

Reference 2: *Parliament is the largest body of the country.*

Reference 3: *Parliament is the highest body of the nation.*

As depicted in Table 6.1, Candidate 1 shares more n-grams with the references, while Candidate 2 shares less words with the references. Candidate 1 shares the biggest n-gram (i.e. '*Parliament is the highest body of')* with the reference 3, and shares '*is the highest body of*' with Reference 1. For a native speaker, it is clear that Candidate 1 is a

better translation and for a machine, it can be detected by choosing the candidate with the higher number of matched n-grams with the references. Although, Candidate 2 shares phrases with all the references, it shares smaller phrases with the references.

Table 6.1: N-gram Matches in Example 6.2

Candidate	Phrase	Reference
Candidate 1	*Parliament is the highest body of*	Reference 3
Candidate 1	*Parliament is the*	References 2
Candidate 1	*is the highest body of*	Reference 1
Candidate 1	*Nation*	References 1 Reference 3
Candidate 1	*body of a*	References 2
Candidate 2	*Assembly is the*	Reference 1
Candidate 2	*is the largest*	References 2
Candidate 2	*the country*	References 2
Candidate 2	*of the*	References1 References2 References 3
Candidate 2	*is the*	References 3

By comparing matched n-grams of candidates with the references, the best candidate can be selected easily. The aim of BLEU metric is to count the matched n-grams by comparing the n-grams of candidates and n-grams of the references. The BLEU calculation procedure starts with comparing unigrams, but the unigrams have the lowest value compare to other n-grams in calculating the BLEU score (Papineni et al., 2002).

6.2.2 Modified N-gram Precision

The fundamental of this method is the precision calculation. To measure the precision, the number of candidate words (i.e. unigram) which appears in any references can be counted and divided by the total number of the candidate's words (Papineni et al., 2002). However, candidates with repetitive words can mislead to high accuracy as depicted in Example 6.3. To avoid obtaining 8/8 accuracy for Example 6.3, reference word should be removed after it is matched with one of the candidates' words.

Example 6.3:

The translations of a Malay language sentence (i.e. *'Berikut adalah contohnya'*):

Candidate: *is is is is is is is.*

Reference 1: *here is an example.*

Reference 2: *this is an example.*

To calculate n-gram precision, first, all the possible phrases that have more than one in the candidate generated, then, phrases will be searched in the references one by one. If a n-gram matched with a reference, the n-gram should be removed from the reference. Finally, the total number of matched n-grams should be divided by the total number of n-grams in candidate (Papineni et al., 2002). In Example 6.2, Candidate 1 achieves a modified tri-gram precision of *6/8*, whereas the lower quality Candidate 2 achieves a modified tri-gram precision of *3/8*. In Example 6.3, the candidate achieves a modified bi-gram precision of 0. The modified n-gram metric can evaluate the adequacy and fluency. The candidates with a higher score of unigram signifies better adequacy, and, the candidates with a higher score of n-grams signify better fluency (Papineni et al., 2002).

6.2.3 BLEU Calculation

Papineni et al. (2002) defined the BLEU score as multiply of the modified precision scores by an exponential brevity penalty factor, where:

p_n: geometric average of the modified n-gram precisions.

$w_{n:}$ positive weights of n-grams up to length of N

c: the length of the candidate translation

r: the effective reference corpus length.

BP: the brevity penalty is computed using Eq. 6.1 and 6.2 (Papineni et al., 2002):

$$BP = \begin{cases} 1 & if\ c > r \\ e^{(1-r/c)} & if \leq r \end{cases} \quad \text{(Eq. 6.1)}$$

Then,

$$BLEU = BP.exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad \text{(Eq. 6.2)}$$

The log domain assists to calculate ranking as shown in Eq. 6.3 (Papineni et al., 2002):

$$\log BLEU = \min\left(1 - \frac{r}{c}, 0\right) + \sum_{n=1}^{N} w_n \log P_n \quad \text{(Eq. 6.3)}$$

6.3 Architecture Evaluation

6.3.1 Implementation

A prototype of the proposed architecture is built using Python programming language. The prototype uses NLTK (Loper & Bird, 2002) as its NLP backbone library. Integrated Development and Learning Environment (IDLE) are used, which is a lightweight and robust Python IDE. Linux-based operating system is preferred while programming with Python. Ubuntu, which is a Debian-based Linux, is installed on an ordinary workstation (i.e. Intel i7-2640M CPU) to implement the prototype. Figure 6.1 and 6.2 display the normalizer GUI, and Table 6.2 and APPENDIX D display a few sample results.

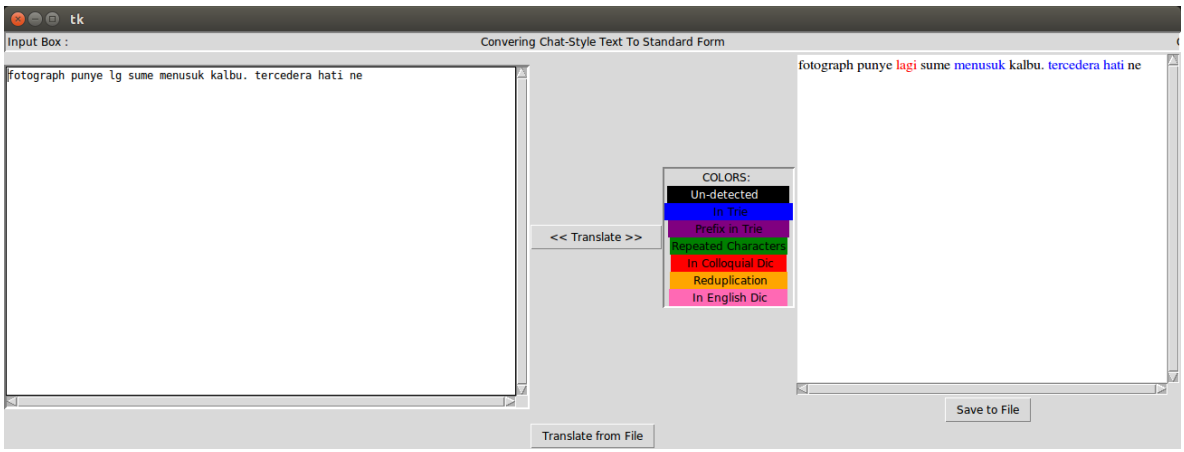

Figure 6.1: GUI Screenshot

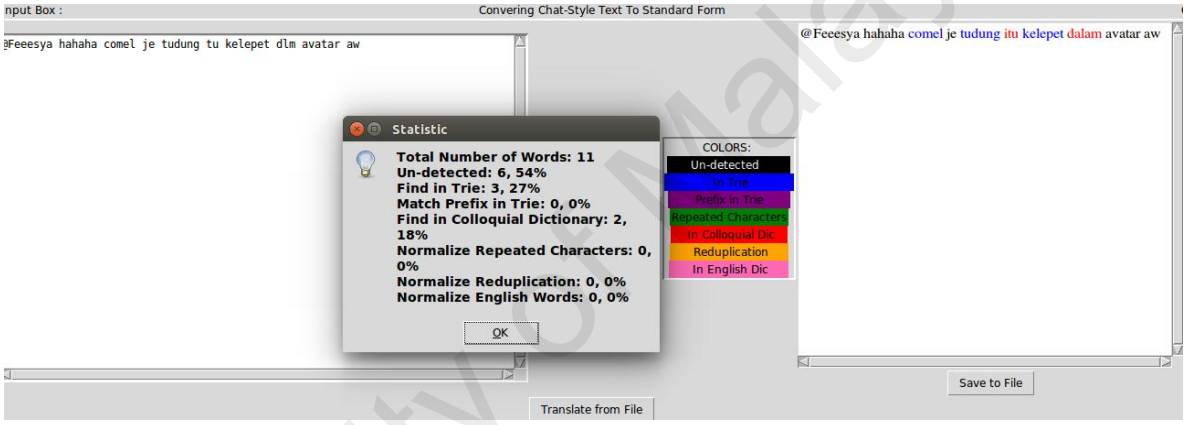

Figure 6.2: GUI Screenshot

Table 6-2: Sample Results

No.	Input/output Screenshots	Result
1.	exam, distracted gila! Ku dah renew sim ku ku still guna no sama ni @_FarisEd_ thank youuu babyyyyy	IV detection English dic
	exam, distracted gila! aku sudah renew sim aku aku masih guna nombor sama ini @_FarisEd_ thank you baby	Colloquial dic. Repeated letters
2.	@ bellzpowerz terimah kasih awakkkk	Repeated letters
	@bellzpowerz terimah kasih awak	IV detection
3.	@FaraFatinah tu lah… okaayy :)	Colloquial dic.
	@FaraFatinah itu lah… okay :)	Repeated letters

6.3.2 Evaluation Results of Proposed Normalizer

Since the first 7500 Tweets of MCC are used in the developing the architecture (i.e. colloquial dictionary), another 1500 Tweets are used for testing the architecture. Two

linguists from University of Malaya converted and verified the conversion of 1500 Tweets to standard Malay. Since MCC is partially aligned by word and message, and not by sentence, the message-aligned part was used to test the architecture.

The architecture is evaluated with its each module to investigate the performance of each module independently. Table 6.3 refers to the evaluation results. The baseline result refers to the similarity between the noisy text (i.e. input text) and the human translated results. In other words, the baseline refers to the correlation between gold standard (human translation) and raw Tweets. The evaluation proves that the architecture can raise the BLEU score by 0.45 (i.e. from 0.46 to 0.91). By using all the modules, 0.91 BLEU score is produced. If the context-aware colloquial dictionary be disabled in the architecture, BLEU score will drop to 0.81. Table 6.3 proves that all the normalization modules contributes to produce the best BLEU score (i.e. 0.91).

Table 6.3: Evaluation results

No.	Mode	Description	BLEU Score
1.	Baseline		0.46
2.	Tokenizer+IV search+Colloquial dic+repeated letter +abbreviation +English dic+De-tokenizer	All modules	0.91
3.	Tokenizer+IV search +repeated letter+abbreviation+English dic+De-tokenizer	Without Colloquial dic	0.80
4.	Tokenizer+IV search+Colloquial dic +abbreviation+English dic+De-tokenizer	Without repeated letter	0.82
5.	Tokenizer+IV search+Colloquial dictionary+repeated letter+English dic+De-tokenizer	Without abbreviation	0.85
6.	Tokenizer+IV search+Colloquial dic+repeated letter +abbreviation +De-tokenizer	Without English dc	0.84

The architecture is examined with a variety of threshold for prefix search in the second modules as shown in Table 6.4. From the table, it shows that the prefix search lessens the accuracy of the system, and the second module works well without prefix search.

The reason behind that is the limited size of Bahasa Wordnet, and alteration at the end of words. For example, if threshold set to 7, and input token is *terbaiknie*, and the word *terbaik* (best) exists in the Trie, the second module will mistakenly distinguish it as an IV word, whilst the correct word is *terbaiknya.*

Table 6.4: Accuracy of Architecture

Mode	BLEU Score
Baseline	0.46
IV search with 3 characters Threshold	0.86
IV search with 4 characters Threshold	0.83
IV search with 5 characters Threshold	0.89
IV search with 6 characters Threshold	0.84
IV search with 7 characters Threshold	0.88
IV without prefix search	0.91

6.4 SMT-like Evaluation

6.4.1 Statistical Machine Translation (SMT)

In SMT, sentences in one language (e.g. Malay) is automatically mapped into another language (e.g. English). The first language is known as the source and the second language is known as the target. There are many forms of SMT approaches such as string-to-string, tree-to-strings, and tree-to-tree. The SMT models can be trained by parallel corpora or monolingual corpora (Osborne, 2017).

The SMT can be formulated by source channel approach (Osborne, 2017):

$$t = argmax_t \left(\sum_i f_i(t, s) \lambda_i \right) \quad \text{(Eq. 6.4)}$$

To discover the most probable target sentence *t* for source sentence *s*, three main stage should be followed (Osborne, 2017):

- *P(s | t):* to assign probability value to the possible sets of soure and target phrases. The translation model also assigns probabilities to these translations, representing their relative correctness.

- *P(e):* to model the general probability of the proposed target text. It is called language model (LM). LM assigns distributions frequency target phrase, that higher score will be assigned to more fluent and common used phrases. Language models are usually smoothed n-gram models, typically conditioning on two (or more) previous words when predicting the probability of the current word.

- argmax opertion: To searach the domain of target sentences to find the highest (i.e. maximized) probility.

In Eq. 6.4, feature function $f_i(t, s)$ calculates $P(s \mid t)$ and $P(t)$, where each of them multiply by weight λi. The weights should be optimized to obtain the best translations with high accuracy. The first step of SMT is to break down target sentences to phrases. Then, the modeling task (i.e. $P(s \mid t)$) determines the break down of source sentence, and mappes source phrases to target system (Osborne, 2017). Table 6.5 refers to an example of English-Malay sentence pair. Table 6.6 shows that source sentence has broken down and paired to target phrases. There are two advantages of phrase-based models compared to word-based models: 1) the phrase-based models can obtain local word order (i.e. order of a word inside a phrase), 2) the phrase-based models make fewer mapping decisions, which cause fewer errors (Osborne, 2017). An important task of SMT systems is to handle phrasal reordering (Osborne, 2017). In translation between to different languages, source and target phrases usually do not follow the same order, however, in the normalization field (i.e. translating noisy text to standard text), the reordering problem is negligible.

Table 6.5: Sentence Pair

Malay	*Bahawa kanak-kanak kecil dan kakaknya memecahkan pintu.*
English	*That little child and his sister break the door.*

Table 6.6: Phrase Pairs

Malay	**English**
Bahawa kanak-kanak kecil	That little child
kanak-kanak kecil dan	little child and
kanak-kanak dan	child and his
dan kakaknya	and his sister
kakaknya memecahkan	his sister break
kakak memecahkan	sister break the
memecahkan pintu.	break the door.

6.4.2 SMT-like Normalization System

Typically, an SMT system translates a sentence from one language to another. An alignment step learns a mapping of words and phrases between the two languages using a training corpus of parallel sentences. During testing, this mapping is used along with language models to translate a sentence from one language to another. While researchers have successfully used SMT method to normalize abbreviations (Aw et al., 2006; Bangalore, Murdock, & Riccardi, 2002; Kobus et al., 2008), the only drawback is the new words handling that leads to poor accuracy in a domain where new phrases are used rapidly.

Systems for text normalization are constructed based on SMT models and parallel corpora. The noisy channel (refer to Eq. 6.4) usually will be generated for converting non-normalized into normalized text. For rapid development of normalization systems at low costs, Schlippe et al. (2010) constructed the parallel corpora with the support of Internet users. Schlippe et al. (2010) normalize text displayed in a web interface, thus providing a parallel corpus of normalized and non-normalized text.

6.4.3 Evaluation Results of SMT Normalizer

The experiment was done by using Moses (Koehn et al., 2007), which is an open-source public domain packages for statistical machine translation. To perform automatic word alignment between noisy and standard text, Giza++ (Och & Ney, 2003) is utilized. The SMT-like system was evaluated with both words-aligned and message-aligned parallel datasets. Section 4.2.4.1 explains that our linguists manully built parallel dataset of Tweets and their corresponding normalized (i.e cleansed) text. However, higher accuracy achieved with the no usage of word alignment. Although Moses allows setting the distortion limit between 0 and 7 for reordering phrases, our experiment without reordering phrases produced better results because Malay language is converted to Malay not another language.

Cleansed part (i.e. reference text) of the testing dataset, which contains 108,373 words, was fed into SRILM (Stolcke, 2002) to build trigram Language Model (LM) with Kneser-Ney smoothing (Kneser & Ney, 1995). Kaufmann and Kalita (2010) asserted that utilizing cleaned Twitter messages instead of conventional corpus for compiling LM can boost the accuracy of the system. Section 6.4.1 explains that the optimum weight (i.e. λ) should be calculated in the noisy channel. A series of weights from 0.0 to 1.0 have been examined to find optimum value of λ. It was found that if the LM has a weight of 0.7, the BLEU score achieves its highest: ~0.8113. Table 6.6 shows the final results of the SMT-like system. Section 2.4.2.1 explains that 9000 Tweets manually cleansed to be used as for training and testing. The dataset is divided into 6 equal sets (i.e. 1500 Tweets) to have 6-fold cross validation. In k-fold cross-validation, the dataset is randomly partitioned into k equal size subsets. Out of the k subsets, a single subset is reserved as the validation data for testing, and the remaining k-1 subsets are used for training. The cross-validation process is then repeated k times (the folds), with each of the k subsets are used once as the testing data. As depicted in

Table 6.7, the k results from the folds are averaged to achieve a single evaluation score, which is 0.8113.

Table 6.7: Accuracy of SMT-like System

6-fold cross validation	BLEU score 3-gram
fold 1	0.8070
fold 2	0.8205
fold 3	0.7982
fold 4	0.8191
fold 5	0.8101
fold 6	0.8129
Average	0.8113

6.5 Discussion and Comparison

The proposed architecture and SMT-like system attain 0.91 and 0.81 BLEU scores respectively. This result proves that a normalization system, which is constructed based on exhaustive analysis, can outperform probabilistic systems that avail machine learning methods. However, several limitations are detected in our proposed approach by analyzing the output of the system. The most important one is that the context-support colloquial dictionary will fail, when the text become very noisy, that is, the preceding and following tokens are misspelled. Another shortcoming is that the English translation module can only convert the correct English words, while users also misspell the English words.

Although the focus of the proposed normalization architecture is to handle Malay language Tweets, approximate comparison can be derived by considering BLEU scores from different normalization studies. Another obstacle in comparison is that a few of the normalization studies only mention achieved BLEU score, and neglect baseline BLEU score, while BLEU score variance between before-normalization and after-normalization shows the accuracy of the approach. In addition, slight distinctions in BLEU score are more meaningful when scores are low (Papineni et al., 2002). Therefore, the difference between 0.30 and 0.32 BLEU scores is much more noticeable

than the difference between 0.80 and 0.82 scores. Table 6.8 shows that our architecture can acquire more than *90%* increase in the BLEU score. The achieved BLEU score par excellence shows that the architecture possesses a reasonable level of competence.

Table 6.8: Accuracy Comparison

Work	Test Dataset	Approach	Baseline BLEU	Normalization BLEU
The proposed approach	1500 Malay Tweets	Multi Module Architecture	0.46	0.91
(Aw et al., 2006)	5000 English SMS	Phrase based SMT	0.5784	0.807
(Kobus et al., 2008)	3000 French SMS	SMT+ ASR	-	0.8
(Beaufort et al., 2010)	30,000 French SMS	Rule-based + Noisy Channel metaphor	-	0.83
(Kaufmann & Kalita, 2010)	1150 English Tweets	Syntactic disambiguation + SMT	0.6799	0.7985
(Gadde et al., 2011)	1000 English SMS	Artificially generated noisy text + SMT	0.448	0.710
(Han & Baldwin, 2011)	549 English Tweets.	Dictionary Lookup +Word Similarity + Context Support	-	0.934
(Lopez Ludeña et al., 2012)	671 Spanish sentences	SMT (architecture for only expanding abbreviations)	-	0.961

The experimental result in Han and Baldwin (2011) shows that Tweet dataset has obtained higher BLEU score that SMS messages. Kaufmann and Kalita (2010) indicate that the baseline (i.e.before normalization) Tweet BLEU score is higher than SMS, therefore, Twitter messages have less OOV tokens compared to SMS. However, Kaufmann and Kalita (2010) assert that the normalizing Twitter message is more complicated due to its irregular pattern of errors, that is, Tweets include higher number of abbreviation styles. Section 4.3.2 explains eight types of abbreviation types in Malay Tweets. Although, Lopez Ludeña et al. (2012) and Han and Baldwin (2011)

produced high BLEU scores (i.e. 0.96 and 0.93), the baseline (before normalization) BLEU score is not reported in their research publication, thus, it is not clear how much the noisy text improved. Nonetheless, our baseline BLEU score and the unknown word analysis prove that Malay language Tweets are noisy to a great extent.

6.6 Summary

BLEU become acceptable evaluation metric in the normalization researches. The proposed architecture and SMT-like systems has been evaluated with the same real data (part of MCC corpus) to discover their accuracy in term of BLEU score. BLEU measures the efforts needed to convert system translation to human translation. The evaluation proves that the architecture can raise the BLEU score from 0.46 to 0.91. The proposed normalizer achieves a better BLEU score comparing to other existing probabilistic and rule-based systems in Table 6.7.

CHAPTER 7: CONCLUSION

7.1 Introduction

Social Network Services (SNSs) and microblogs are increasingly popular forms of communications. It is a potent and cost-effective platform that facilitates business and social communication. Many SNS applications are developed to provide services that add value to the business processes and promote social interaction within the communities. SNSs and microblogs explosion has drawn researchers' attention to this research area like opinion mining and sentiment analyzing, to name a few (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011; Diakopoulos & Shamma, 2010; Thelwall, Buckley, Paltoglou, Cai, & Kappas, 2010; X. Zhang, Fuehres, & Gloor, 2012). However, one of the biggest obstacles to apply IR methods to UGC is the existence of noisy text. Most of NLP applications and text mining algorithms are designed to apply to clean texts. Along with more and more demand for UGC processing, the research in normalization, the process of cleaning noisy text, has become an increasingly important topic.

A large portion of social media content does not obey grammatical and orthographic rules, and many people are accustomed to bad grammar in SNS sphere. Even people who have been speaking English for a long time tend to make common grammatical mistakes such as verb tense errors, subject/verb agreement errors, noun/pronoun errors, double negatives, sentence fragments, and run-on sentences. It is found that languages have common grammatical errors such as sentence fragments, and a variety of orthographic rules are common between Latin alphabet writing systems. The lack of punctuation is a typical phenomenon in social media content. In addition, there are an enormous number of typographic errors and misspelled words in SNS context. There are two major categories of misspelled words: these are abbreviations and

interjections. Normally, users shorten (abbreviate) words by phonetic spelling to decrease keystroke such as "*gr8*" for "*great*", and "*wnt*" for "*went*". The interjection misspelling denotes emotions or emphasis by using majuscule letters or repeating letters such as "*WeLL*" for "*well*", and "*hiiiii*" for "*hi*". This study only discusses normalization of misspelled words on Twitter and does not cover all possible errors like typos and grammatical errors.

Twitter, with over 500 million registered users, is the most popular Microblog in 2012 (Dugan, 2012). Twitter users send text messages known as "Tweet" consisting of a maximum of 140 characters. Twitter coined its own terminology such as "Retweet" and "RT". Apart from general features of chatspeak text, Tweet has unique characteristics, including ample use of at-sign (@) and number-sign (#). The username and Tweet's topic can be distinguished by their specific syntax using the pattern @username and #topic (e.g. *@John*, and *#malaria*). Therefore, it is important to explore Tweet's features in normalizing them, as it has been done in this research to normalize the Malay language Tweets.

Malay language is the fourth leading language that practices over the Twitter (Guyot, 2010). It motivates us to propose an approach for Malay language Tweets normalization. Malay, with more than 200 million speakers, is the sixth most spoken in the world (Teeuw, 1959). However, incomplete or unavailable digital resources for the language (Ehsan, Quah, et al., 2001; Kassim, 2008; Noor et al., 2011) lead us to put the Malay language in less studied language category from a computational point of view. For example, to the best of our knowledge, there is none study on the Malay spell checker and Named Entity Recognition (NER). The limited resource of Malay language inspired us to follow a specific methodology to build a normalization system. The methodology articulates three major phases:

I. Performing analyses to scrutinize features and characteristics of colloquial Malay.

II. Designing and implementing normalization architecture based on the result of the analyses.

III. Evaluating the system and comparing it with comparison baseline (i.e. SMT normalizer) approach.

7.2 Overview of Research

This section reviews the main parts of this research work by highlighting the output of the work which corresponds to each of the objectives discussed in Chapter 1.

Objective 1. To compile a corpus that represents the colloquial Malay language in Twitter.

A corpus is developed to represent the colloquial Malay language, which is named Malay Chat-style-text Corpus (MCC). To guaranty the representativeness of MCC, sampling standards, a variety of texts, and chronology are investigated. MCC contains a concordance that undertakes corpus expandability and annotatability. The MCC, which encloses 1 million Tweets, includes 14,484,384 word instances, 646,807 unique vocabularies, and metadata, such as used Twitter client application, posting time, and type of Twitter message (i.e. simple Tweet, Retweet, and Reply).

Objective 2. To analyze colloquial language and standard language corpora

Malay unknown words are identified and their features, including their frequencies and patterns of abbreviations are extracted. APPENDIX A list out the most 200 frequent words in Malay language Twitter messages. The most commonly detected OOV words in the MCC are English words, while abbreviations and interjections are second and third, respectively. Eight types of abbreviation pattern are detected consisting of

reduplication, vowel elimination, '*r*' elimination, affixes elimination, initial letter elimination, last letter elimination, and word combinations.

The study presents a method to eliminate extra repeated letters from Malay words, based on the patterns of repeated letters in the standard Malay language. To find morphological features of Malay word, an analysis was performed over DBP corpus, consisting of 135 million words. The result of the analysis shows that repetition of same letter does not occur in Malay words except for nine certain conditions, which is elaborated in Section 4.3.6.

Objective 3. To design a Malay language Tweet normalization architecture which can convert the Malay language Tweet lingo into its standard Malay language.

Malay language Tweet normalization architecture is designed to convert the Malay language Tweet lingo to its standard Malay language. The design of the architecture is based on the result of objective 2. The architecture comprises seven modules:

i. Enhanced tokenization

ii. IV words detection

iii. Colloquial dictionary lookup

iv. Repeated letters elimination

v. Twitter abbreviation normalizer

vi. English word translation

vii. De-tokenization.

After tokenization, OOV words are detected and send to normalization modules. The token manipulation is tracked and recorded through their life cycle using tags. For

example, IW tags are added to correct words at the beginning steps to prevent them from manipulation in the normalization modules.

Algorithms are formulated for the normalization modules based on the results of the Objectives 2. IV words detection, English word translation, and colloquial dictionary lookup basically rely on robust data structure and do not need complex algorithms. Three algorithms are proposed they are:

- Enhanced Malay chat-style text tokenization and detokenization,
- Extra repeated letter elimination, and
- Abbreviation normalization.

Objective 4. To evaluate the performance of the Malay language Tweet normalization architecture with our benchmark.

The performance of Malay language Tweet normalization architecture is evaluated. The cross-fold evaluation proves that the architecture can raise the BLEU score by 0.45 from 0.46 to 0.91. The comparison of results with SMT approach shows that the architecture possesses a reasonable level of competence.

7.3 Contribution

The aim of this research is to design a Malay language Tweet normalizer. Similarly, the main contribution of this work is the Malay language Tweet normalizer architecture. A normalizer, to convert Malay OOV words to standard form of language, is designed, implemented and evaluated. It is designed based on the features of the Malay language. Thus, there are two main contributions:

Contribution 1. MCC corpus, which consists of 1 million Malay language Tweets, 14,484,384 word instances, 646,807 unique vocabularies, and their metadata, is provided. Sampling standards and criteria for corpus creation are

investigated and practiced to compile a corpus that represents a colloquial Malay language. MCC can be used for corpus-driven analysis as performed in this research, where, section 4.3 elaborates the frequency of OOV words and identified patterns of abbreviations.

Contribution 2. Malay language Twitter message normalization architecture is proposed, which is composed of seven modules: 1) enhanced tokenized, 2) in-vocabulary identifier, 3) context aware colloquial dictionary, 4) repeated letter eliminator, 5) abbreviation normalizer, 6) English dictionary, and 7) de-tokenizer. The normalizer can convert the noisy text to standard language with high accuracy, where, 0.91 BLEU score is reported. The normalizer is designed and formulated based on the results of the corpus-driven analysis. There are many sophisticated features in the normalizer such as a context aware colloquial dictionary. Since the dictionary store preceding and following words of the OOV, it can detect the best translation based on the context. Table 6.3 shows that without colloquial dictionary the BLEU score drops to 0.80.

7.4 Conclusion

Recent years have witnessed the explosive growth of online Social Network Services (SNSs) like Twitter, with nearly 400 million Tweets per day, is the most used and well-known worldwide microblogging service (Otsuka, Wallace, & Chiu, 2014). It's the sheer volume of messages on Twitter that causes Tweets became a valuable resource for researchers. However, most of NLP and Text Mining methods are constructed to apply to standard text and applying them to a text with a variety of colloquialisms, such as Twitter messages, may produce inaccurate results. Thus, to work with Twitter messages, normalizing the noisy text is essential. The aim of this

study is to normalize Malay language Tweets, the fourth most used language in Twitter.

In this research, normalization architecture is designed based on features of colloquial and standard Malay. To extract the characteristics of normal and chat-style Malay, four corpus-driven analyses have been done:

1) Analyzing the frequency distribution of unknown words indicates the most prevalent types of OOV words.

2) Analyzing abbreviation pattern demonstrates that it is a straightforward task to detect and correct a few given types of word shortening.

3) Analyzing consecutive repetition of letters in Malay morphology exhibits how to tackle the sequence of extra repeated letters in a row.

4) Inspecting status of periods and majuscule letters in Malay language Tweets demonstrates how to deal with them in the tokenization level.

The first analysis revealed that English terms are the most prevalent unknown words, followed by abbreviated words and letter repetition. The result of the second analysis shows that users follow certain methods for a few types of abbreviation, i.e. abbreviating reduplication and negation. The outcome of the third analysis unveiled that there is no any sequence of repeated letters in a row in Malay morphology. The last analysis acknowledges the fact that most Twitter sentences do not end with a period and do not begin with a majuscule letter.

The normalization architecture includes seven modules in a pipeline workflow. The first one is the enhanced tokenization module, which is designed based on attributes of Malay language Tweets. This module also can detect certain types of proper noun. The second module concerned with distinguishing IV words to protect them against

alteration in the next modules. The third module is the translating tokens using a colloquial dictionary. A Malay colloquial dictionary is compiled including 765 entries. The fourth module is the elimination of extra repeated letters, then, followed by abbreviated words correction module, which is the fifth module. The sixth module is the English word translation module and the de-tokenization is the final module.

The architecture is implemented using Python programming language and its accuracy was measured in term of BLEU score. The system is tested over 1500 parallel Malay language Tweets. According to the experimental results, the proposed architecture increases BLEU score from 0.46 (before normalization) to 0.91. For the sake of comparison, SMT-like normalization system is implemented and evaluated. The SMT-like normalization, which considers the noisy text as a source language and the standard text as a target language, is a probabilistic approach. The phrase-translation table is generated by using Moses (Koehn et al., 2007), and a trigram LM is produced by using SRILM (Stolcke, 2002). With chosen golden word alignment in GIZA++ (Och & Ney, 2003), when it is trained and tested over 6-fold the cross validation by using 9000 parallel Malay language Tweets, the 0.81 BLEU score is achieved. As a conclusion, the evaluation of the approach shows that it is capable of normalizing Malay language Tweets with acceptable accuracy.

7.5 Future Work

It is widely accepted that a spell checker can improve performance and accuracy of normalization systems (Liu, Weng, & Jiang, 2012; Liu et al., 2011; Xue et al., 2011) whereas, to the best of our knowledge, there is no study on the Malay conventional spell checker. As future work, the architecture might be integrated with the Malay spell correction system to improve accuracy.

Another limitation of this work is the usage of the prefix search for the dictionary-based IV word detection. There are two main factors that drop the accuracy of the IV word detection module:

- The available Malay dictionaries are not complete and they only include a base form of words.
- Abbreviations may occur at the prefix and postfix of the noisy words. For example, if the input token is *terbaiknie*, and the word *terbaik* (best) exists in the Trie, the module will mistakenly distinguish it as an IV word, while the correct word is *terbaiknya*.

Finally, the limitation of the proposed normalizer is its dependency to the MCC corpus. The normalizer is designed based on the features of the MCC corpus. The MCC represents the colloquial languages that practiced in Malay Twitter messages during January 2012 till November 2012. Generally, languages change over time (Graddol, 2004). Thus, the type of abbreviations that would be used in the future is different from the detected patterns of abbreviation in the MCC. As a future work, a dynamic Twitter corpus must be implemented to detect the recent features of chat-style-text lingo. As a road map, a comparison of the Malay noisy text with other languages helps to deeply understand Malay chat style.

REFERENCES

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment Analysis of Twitter Data. In *Proceedings of the Workshop on Languages in Social Media* (pp. 30–38). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2021109.2021114

Ahmad, M., & Mathkour, H. (2009). A Pattern Matching Approach for Redundancy Detection in Bi-lingual and Mono-lingual Corpora. In *Proceedings of International MultiConference of Engineers & Computer Scientists* (pp. 526–531). Hong Kong: International Association of Engineers.

Aizawa, A. (2003). An information-theoretic perspective of tf-idf measures. *Inf. Process. Manage.*, *39*(1), 45–65. https://doi.org/10.1016/S0306-4573(02)00021-3

Arshi Saloot, M., Idris, N., Shuib, L., Gopal Raj, R., & Aw, A. (2015). Toward Tweets Normalization Using Maximum Entropy. In *Proceedings of the Workshop on Noisy User-generated Text* (pp. 19–27). Beijing, China: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W15-4303

Atkins, S., Clear, J., & Ostler, N. (1992). Corpus Design Criteria. *Literary and Linguistic Computing*, *7*(1), 1–16. https://doi.org/10.1093/llc/7.1.1

Aw, A. T., & Lee, L. H. (2012). Personalized Normalization for a Multilingual Chat System. In *Proceedings of the ACL 2012 System Demonstrations* (pp. 31–36). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2390470.2390476

Aw, A., Zhang, M., Xiao, J., & Su, J. (2006). A Phrase-based Statistical Model for SMS Text Normalization. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions* (pp. 33–40). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1273073.1273078

Bakar, H. A. (2009). Code-switching in Kuala Lumpur Malay: The "Rojak" Phenomenon. *EXPLORATIONS a Graduate Student Journal of Southeast Asian Studies*, *9*, 99–107.

Baldwin, T., & Li, Y. (2015). An In-depth Analysis of the Effect of Text Normalization in Social Media. In R. Mihalcea, J. Y. Chai, & A. Sarkar (Eds.), *{NAACL} {HLT} 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015* (pp. 420–429). The Association for Computational Linguistics. Retrieved from http://aclweb.org/anthology/N/N15/N15-1045.pdf

Bang, M. (2009). *Representation of foreign countries in the US press : a corpus study*. Retrieved from http://etheses.bham.ac.uk/902/

Bangalore, S., Murdock, V., & Riccardi, G. (2002). Bootstrapping Bilingual Data Using

Consensus Translation for a Multilingual Instant Messaging System. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1* (pp. 1–7). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/1072228.1072362

Basri, S. B., Alfred, R., & On, C. K. (2012). Automatic spell checker for Malay blog. In *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on* (pp. 506–510). https://doi.org/10.1109/ICCSCE.2012.6487198

Beaufort, R., Roekhaut, S., Cougnon, L.-A., & Fairon, C. (2010). A Hybrid Rule/Model-based Finite-state Framework for Normalizing SMS Messages. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 770–779). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1858681.1858760

Biber, D. (1993). Representativeness in Corpus Design. *Literary and Linguistic Computing*, *8*(4), 243–257. https://doi.org/10.1093/llc/8.4.243

Biber, D. (2012). Corpus-Based and Corpus-driven Analyses of Language Variation and Use. In *The Oxford Handbook of Linguistic Analysis*. Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199544004.013.0008

Bieswanger, M. (2007). 2 abbrevi8 or not 2 abbrevi8: A Contrastive Analysis of Different Space- and Time-Saving Strategies in English and German Text Messages. *Texas Linguistic Forum, Vol. 50*.

Boguraev, B. (1990). Review of "Looking Up: An Account of the COBUILD Project in Lexical Computing" by John M. Sinclair. Collins ELT 1987. *Comput. Linguist.*, *16*(3), 184–186. Retrieved from http://dl.acm.org/citation.cfm?id=98377.976196

Bouma, G. (2009). Normalized (Pointwise) Mutual Information in Collocation Extraction. Retrieved from http://www.google.de/url?sa=t&rct=j&q=normalized (pointwise) mutual informationin collocation extraction&source=web&cd=2&cad=rja&ved=0CE4QFjAB&url=https://svn.spraakdata.gu.se/repos/gerlof/pub/www/Docs/npmi-pfd.pdf&ei=prr5UNWSBs_TsgaPzoD4Bg&usg=AFQjCNFAHJHKG5tLXCNmGJw4yRqX2WuPuA&bvm=bv.41248874,d.Yms

Callison-Burch, C., Osborne, M., & Koehn, P. (2006). Re-evaluating the Role of BLEU in Machine Translation Research. In *In EACL* (pp. 249–256).

Carter, R., & McCarthy, M. (2006). *Cambridge grammar of English: a comprehensive guide : spoken and written English grammar and usage.* Cambridge: Cambridge University Press.

Choudhury, M., Lin, Y.-R., Sundaram, H., Candan, K. S., Xie, L., & Kelliher, A. (2010). How Does the Data Sampling Strategy Impact the Discovery of Information Diffusion in Social Media? In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*. Retrieved from http://www.public.asu.edu/~mdechoud/pubs/icwsm_10.pdf

Choudhury, M., Saraf, R., Jain, V., Mukherjee, A., Sarkar, S., & Basu, A. (2007).

Investigation and Modeling of the Structure of Texting Language. *Int. J. Doc. Anal. Recognit.*, *10*(3), 157–174. https://doi.org/10.1007/s10032-007-0054-0

Clark, A. (2003). Pre-processing very noisy text. In *Proceedings of Workshop on Shallow Processing of Large Corpora, March 27 (SProLaC03)* (pp. 12–22). Lancaster, UK: UCREL. Retrieved from http://www.issco.unige.ch/en/staff/clark/SprolacPaper.doc.pdf

Clark, E., & Araki, K. (2011). Text Normalization in Social Media: Progress, Problems and Applications for a Pre-Processing System of Casual English. *Procedia - Social and Behavioral Sciences*, *27*(0), 2–11. https://doi.org/http://dx.doi.org/10.1016/j.sbspro.2011.10.577

Contractor, D., Faruquie, T. A., & Subramaniam, L. V. (2010). Unsupervised Cleansing of Noisy Text. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters* (pp. 189–196). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1944566.1944588

Cook, P., & Stevenson, S. (2009). An Unsupervised Model for Text Message Normalization. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity* (pp. 71–78). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1642011.1642021

Dahlmann, I. (2007). How Big is Big Enough? Methodological Considerations on the Determination of Corpus Size for the Study of Frequent Multi-Word Units (MWUs) in Spoken Language. In *Corpus Linguistics*. Birmingham.

Diakopoulos, N. A., & Shamma, D. A. (2010). Characterizing Debate Performance via Aggregated Twitter Sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1195–1198). New York, NY, USA: ACM. https://doi.org/10.1145/1753326.1753504

Dugan, L. (2012). Twitter To Surpass 500 Million Registered Users On Wednesday. Retrieved from https://www.adweek.com/digital/500-million-registered-users/

Ehsan, S. D., Quah, C. K., Bond, F., & Yamazaki, T. (2001). Design and Construction of a Machine-Tractable Malay-English Lexicon. In *In Asialex-2001 , Seoul* (pp. 200–205).

Fairon, C., & Paumier, S. (2006). A translated corpus of 30,000 French SMS. In *LREC* (p. 10pp.). G{ê}nes, Italy. Retrieved from https://hal-upec-upem.archives-ouvertes.fr/hal-00621421

Gadde, P., Goutam, R., Shah, R., Bayyarapu, H. S., & Subramaniam, L. V. (2011). Experiments with Artificially Generated Noise for Cleansing Noisy Text. In *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data* (p. 4:1--4:8). New York, NY, USA: ACM. https://doi.org/10.1145/2034617.2034622

Ghadessy, M., Henry, A., & Roseberry, R. L. (2001). *Small Corpus Studies and ELT:*

Theory and Practice. Amesterdam: John Benjamins.

Graddol, D. (2004). The Future of Language. *Science, 303*(5662), 1329–1331. https://doi.org/10.1126/science.1096546

Groom, N. (2009). Phraseology and Epistemology in Academic Book Reviews: A Corpus-Driven Analysis of Two Humanities Disciplines. In K. Hyland & G. Diani (Eds.), *Academic Evaluation: Review Genres in University Settings* (pp. 122–139). London: Palgrave Macmillan UK. https://doi.org/10.1057/9780230244290_8

Guyot, P. (2010, February 24). Half of messages on Twitter are not in English: Japanese is the second most used language. *Semiocast*. Retrieved from https://semiocast.com/downloads/Semiocast_Half_of_messages_on_Twitter_are_not_in_English_20100224.pdf

Han, B., & Baldwin, T. (2011). Lexical Normalisation of Short Text Messages: Makn Sens a #Twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1* (pp. 368–378). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2002472.2002520

Han, B., Cook, P., & Baldwin, T. (2012). Automatically Constructing a Normalisation Dictionary for Microblogs. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 421–432). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2390948.2391000

Hasund, K. (1998). Explorations in Corpus Linguistics. In A. Renouf (Ed.). Amsterdam & Atlanta: Rodopi.

How, Y., & Kan, M. (2005). Optimizing predictive text entry for short message service on mobile phones. In *in Human Computer Interfaces International (HCII 05). 2005: Las Vegas*.

Hsu, C.-L., & Lin, A. J. (2013). The effect of community identity on continuance intention of microblogging. *International Journal of Electronic Business, 10*(4), 355–382. https://doi.org/10.1504/IJEB.2013.056784

Hunston, S., & Francis, G. (2000). *Pattern Grammar A corpus-driven approach to the lexical grammar of English*. Amesterdam: John Benjamins.

Jehl, L., Hieber, F., & Riezler, S. (2012). Twitter Translation Using Translation-based Cross-lingual Retrieval. In *Proceedings of the Seventh Workshop on Statistical Machine Translation* (pp. 410–421). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2393015.2393074

Jones, R., & Ghani, R. (2000). Automatically Building a Corpus for a Minority Language from the Web. In *Proceedings of the Student Research Workshop at the 38th Annual Meeting of the Association for Computational Linguistics* (pp. 29–36).

Kadir, A. R., Musa, H., Azman, A., & Abdullah, M. T. (2011). Syllabification Algorithm based on Elicitation Method and Syllable Matching Rules for Malay Language. *Journal of Computer Science and Engineering*, *8*(2), 1–7.

Karimah, N., Aziz, M. J. A., Noah, S. A., & Hamzah, M. P. (2011). nya as anaphoric word: A proposed solution. In *Semantic Technology and Information Retrieval (STAIR), 2011 International Conference on* (pp. 249–254).

Kasesniemi, E.-L., & Rautiainen, P. (2002). Perpetual Contact. In J. E. Katz & M. A. Aakhus (Eds.) (pp. 170–192). New York, NY, USA: Cambridge University Press. Retrieved from http://dl.acm.org/citation.cfm?id=644547.644559

Kassim, A. M. (2008). Malay Language As A Foreign Language And The Singapore ' s Education System. *Online Journal of Language Studies*, *8*(1), 47–56.

Kaufmann, M., & Kalita, J. (2010). Syntactic normalization of Twitter messages. *International Conference on Natural Language Processing, Kharagpur, India*.

Kiss, T., & Strunk, J. (2006). Unsupervised Multilingual Sentence Boundary Detection. *Comput. Linguist.*, *32*(4), 485–525.

Klakow, D., & Peters, J. (2002). Testing the correlation of word error rate and perplexity. *Speech Communication*, *38*(1), 19–28.

Kneser, R., & Ney, H. (1995). Improved backing-off for M-gram language modeling. In *International Conference on Acoustics, Speech, and Signal Processing, 1995. ICASSP-95* (Vol. 1, pp. 181–184). Detroit, MI: IEEE Computer Society.

Kobus, C., Yvon, F., & Damnati, G. (2008). Normalizing SMS: Are Two Metaphors Better Than One? In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1* (pp. 441–448). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1599081.1599137

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., … Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (pp. 177–180). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1557769.1557821

Koehn, P., & Monz, C. (2006). Manual and Automatic Evaluation of Machine Translation Between European Languages. In *Proceedings of the Workshop on Statistical Machine Translation* (pp. 102–121). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1654650.1654666

Leech, G., Hundt, M., Mair, C., & Smith, N. (2009). Change in contemporary English: a grammatical study. Cambridge University Press.

Li, C., & Liu, Y. (2012). Improving Text Normalization using Character-Blocks Based Models and System Combination. In *Proceedings of COLING 2012* (pp. 1587–

1602). Mumbai, India: The COLING 2012 Organizing Committee. Retrieved from http://www.aclweb.org/anthology/C12-1097

Ling, W., Dyer, C., Black, A. W., & Trancoso, I. (2013). Paraphrasing 4 Microblog Normalization. In *Conference on Empirical Methods in Natural Language Processing* (pp. 73–84). Seattle, Washington: Association for Computational Linguistics. Retrieved from http://aclweb.org/anthology/D/D13/D13-1008.pdf

Liu, F., Weng, F., & Jiang, X. (2012). A Broad-Coverage Normalization System for Social Media Language. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics Volume 1 Long Papers*, (July), 1035–1044. Retrieved from http://www.aclweb.org/anthology/P12-1109

Liu, F., Weng, F., Wang, B., & Liu, Y. (2011). Insertion, Deletion, or Substitution?: Normalizing Text Messages Without Pre-categorization nor Supervision. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2* (pp. 71–76). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2002736.2002753

Loper, E., & Bird, S. (2002). NLTK: The Natural Language Toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1* (pp. 63–70). Stroudsburg, PA, USA: Association for Computational Linguistics.

Lopez Ludeña, V., San Segundo, R., Montero, J. M., Barra Chicote, R., & Lorenzo, J. (2012). Architecture for Text Normalization using Statistical Machine Translation techniques. In *IberSPEECH 2012* (pp. 112–122). Madrid, Spain: Springer.

Lui, M., & Baldwin, T. (2012). langid.py: an off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations* (pp. 25–30). Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2390470.2390475

Mair, C., Hundt, M., Leech, G. N., & Smith, N. (2002). Short term diachronic shifts in part-of-speech frequencies. *International Journal of Corpus Linguistics*, *7*(2), 245–264.

Manning, C. D., & Schuetze, H. (1999). *Foundations of Statistical Natural Language Processing* (1st ed.). The MIT Press. Retrieved from http://amazon.com/o/ASIN/0262133601/

Mapa, E., Wattaladeniya, L., Chathuranga, C., Dassanayake, S., de Silva, N., Kohomban, U. S., & Maldeniya, D. (2013). Normalization in Social Media by using Spell Correction and Dictionary Based Approach.

Martelli, A. (2003). *Python in a Nutshell*. (P. Ferguson, Ed.). Sebastopol, CA, USA: O'Reilly & Associates, Inc.

Matthiessen, C. (2006). Frequency profiles of some basic grammatical systems: an interim report. In *System and Corpus Exploring Connections*. EQUINOX PUBLISHING.

McEnery, T., & Wilson, A. (2001). *Corpus Linguistics (Edinburgh Textbooks in Empirical Linguistics)* (2nd ed.). Edinburgh University Press. Retrieved from http://amazon.com/o/ASIN/0748611657/

McEnery, T., Xiao, R., & Tono, Y. (2006). *Corpus-Based Language Studies: an Advanced Resource Book*. London: Routledge.

Mohanthy, A., Majhi, kumar S., & Chaupattnaik, S. (2013). Language Normalisation of Noisy Text Data. *International Journal of Advanced Computational Engineering and Networking*, *1*(3), 36–39. Retrieved from http://iraj.in/journal/IJACEN/vol1_issue3/36-39.pdf

Montero, J. M., & Lorenzo, J. (2011). Architecture for Text Normalization using Statistical Machine Translation techniques.

Noor, N. H. B. M., Sapuan, S., & Bond, F. (2011). Creating the Open Wordnet Bahasa. In H. H. Gao & M. Dong (Eds.), *PACLIC* (pp. 255–264). Digital Enhancement of Cognitive Development, Waseda University. Retrieved from http://dblp.uni-trier.de/db/conf/paclic/paclic2011.html#NoorSB11

Och, F. J., & Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. *Comput. Linguist.*, *29*(1), 19–51.

Oliva, J., Serrano, J. I., Del Castillo, M. D., & Igesias, Á. (2013). A SMS Normalization System Integrating Multiple Grammatical Resources. *Natural Language Engineering*, *19*(01), 121–141.

Osborne, M. (2017). Statistical Machine Translation. In *Encyclopedia of Machine Learning and Data Mining* (pp. 1173–1177).

Otsuka, E., Wallace, S. A., & Chiu, D. (2014). Design and Evaluation of a Twitter Hashtag Recommendation System. In *Proceedings of the 18th International Database Engineering & Applications Symposium* (pp. 330–333). New York, NY, USA: ACM.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 311–318). Stroudsburg, PA, USA: Association for Computational Linguistics.

Paran, A. (1993). Review: the lexical syllabus a new approach to language teaching. *ELT Journal*, *47*(4), 363–365.

Partington, A. (2006). *The Linguistics of Laughter: a Corpus-Assisted Study of Laughter- Talk*. London: Routledge.

Pecina, P. (2008). A machine learning approach to multiword expression extraction. Retrieved from http://www.lrec-conf.org/proceedings/lrec2008/workshops/W20_Proceedings.pdf#page=58

Pennell, D. L., & Liu, Y. (2010). Normalization of text messages for text-to-speech.

Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference.

Pennell, D., & Liu, Y. (2011a). A Character-Level Machine Translation Approach for Normalization of SMS Abbreviations. In *Proceedings of 5th International Joint Conference on Natural Language Processing* (pp. 974–982). Chiang Mai, Thailand: Asian Federation of Natural Language Processing. Retrieved from http://www.aclweb.org/anthology/I11-1109

Pennell, D., & Liu, Y. (2011b). Toward text message normalization: Modeling abbreviation generation. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on* (pp. 5364–5367).

Popescu, A.-M., Pennacchiotti, M., & Paranjpe, D. (2011). Extracting Events and Event Descriptions from Twitter. In *Proceedings of the 20th International Conference Companion on World Wide Web* (pp. 105–106). New York, NY, USA: ACM.

Raghunathan, K., & Krawczyk, S. (2018). CS224N : Investigating SMS Text Normalization using Statistical Machine Translation.

Rahim, H. A. (2014). CORPORA IN LANGUAGE RESEARCH IN MALAYSIA. *Kajian Malaysia*, *32*(1), 1–14.

Rayson, P., & Wilson, T. M. (eds. . A. (2003). *A Rainbow of Corpora*. LINCOM publishers. Retrieved from http://amazon.com/o/ASIN/3895868728/

Renouf, A. (2002). The Time Dimension in Modern English Corpus Linguistics. *Language and Computers*, *42*(1), 27–41.

Rivest, R. (1992, April). {RFC 1321}: The {MD5} Message-Digest Algorithm. Retrieved from ftp://ftp.internic.net/rfc/rfc1321.txt,

Saloot, M. A., Idris, N., & Mahmud, R. (2014). An architecture for Malay Tweet normalization. *Information Processing & Management*, *50*(5), 621–633.

Sampson, G. (2003). (eds.), Variation in English: multi-dimensional studies. Studies in Language and Linguistics. In Susan Conrad & D. Biber (Eds.), *English Language and Linguistics* (Vol. 7, pp. 164–167).

Samsudin, N., Puteh, M., Hamdan, A. R., & Nazri, M. Z. A. (2012). Normalization of Common NoisyTerms in Malaysian Online Media. In *Proceedings of the Knowledge Management International Conference* (pp. 515–520). Johor Bahru, Malaysia: UUM Press. Retrieved from http://www.kmice.cms.net.my/ProcKMICe/KMICe2012/PDF/CR204.pdf

Schlippe, T., Zhu, C., Gebhardt, J., & Schultz, T. (2010). Text normalization based on statistical machine translation and internet user support. In T. Kobayashi, K. Hirose, & S. Nakamura (Eds.), *INTERSPEECH* (pp. 1816–1819). ISCA. Retrieved from http://dblp.uni-trier.de/db/conf/interspeech/interspeech2010.html#SchlippeZGS10

Scott, M. (1996). *Wordsmith Tools*. Oxford.

Scott, M. (2001). Comparing corpora and identifying key words, collocations, and frequency distributions through the WordSmith Tools suite of computer programs. *Small Corpus Studies and ELT*, 47–67.

Segerstad, Y. H. af. (2002). *Use and Adaptation of the Written Language to the Conditions of Computer-Mediated Communication*. University of Gothenburg, Sweden.

Sinclair, J. (1991). *Corpus Concordance and Collocation (Describing English Language)*. Oxford Univ Pr (Sd). Retrieved from http://amazon.com/o/ASIN/0194371441/

Sinclair, J. (2004). Intuition and annotation; the discussion continues. *Language and Computers*, *49*(1).

Smith, J., & Padi, P. (2006). Lets make a dictionary. In *Proceedings of the the Eighth Biennial Conference of the Borneo Research Council (BRC)* (pp. 515–520). Sarawak, Malaysia: Borneo Research Council (BRC).

Sproat, R., Black, A. W., Chen, S., Kumar, S., Ostendorf, M., & Richards, C. (2001). Normalization of non-standard words. *Computer Speech & Language*, *15*(3), 287–333.

Stolcke, A. (2002). SRILM-an extensible language modeling toolkit. In *Proceedings International Conference on Spoken Language Processing* (pp. 257–286).

Stubbs, M. (1986). *Educational linguistics*. New York, NY, USA: Oxford.

Tagg, C. (2009, July). *A corpus linguistics study of SMS text messaging*. University of Birmingham. Retrieved from http://etheses.bham.ac.uk/253/

Teeuw, A. (1959). THE HISTORY OF THE MALAY LANGUAGE. *Bijdragen Tot de Taal-, Land- En Volkenkunde*, *115*(2), 138–156. Retrieved from http://www.jstor.org/stable/27860189

Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, *61*(12), 2544–2558.

Thurlow, C., & Brown, A. (2003). Generation Txt? The sociolinguistics of young people's text-messaging.

Tognini-Bonelli, E. (2001). *Corpus linguistics at work*. Amesterdam: John Benjamins.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Election Forecasts With Twitter: How 140 Characters Reflect the Political Landscape. *Social Science Computer Review*.

Vivian, R., Barnes, A., Geer, R., & Wood, D. (2014). The academic journey of university students on Facebook: an analysis of informal academic-related activity over a semester. *Research in Learning Technology*, *22*(0). Retrieved from

http://www.researchinlearningtechnology.net/index.php/rlt/article/view/24681

Wang, A., Kan, M.-Y., Andrade, D., Onishi, T., & Ishikawa, K. (2013). Chinese Informal Word Normalization: an Experimental Study. In *International Joint Conference on Natural Language Processing* (pp. 127–135). Nagoya, Japan: Asian Federation of Natural Language Processing. Retrieved from http://www.comp.nus.edu.sg/~wangaobo/papers/IJCNLP-2013.pdf

Wang, P., & Ng, H. T. (2013). A Beam-Search Decoder for Normalization of Social Media Text with Application to Machine Translation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 471–481). Atlanta, Georgia: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/N13-1050

Wattam, S. M. (2015). *Technological Advances in Corpus Sampling Methodology*. Lancaster University.

Webster, J. J., & Kit, C. (1992). Tokenization As the Initial Phase in NLP. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 4* (pp. 1106–1110). Stroudsburg, PA, USA: Association for Computational Linguistics.

Wei, Z., Zhou, L., Li, B., Wong, K.-F., & Gao, W. (2011). Exploring Tweets Normalization and Query Time Sensitivity for Twitter Search. In E. M. Voorhees & L. P. Buckland (Eds.), *TREC*. National Institute of Standards and Technology (NIST). Retrieved from http://dblp.uni-trier.de/db/conf/trec/trec2011.html#WeiZLWGWE11

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

Xue, Z., Yin, D., & Davison, B. D. (2011). Normalizing Microtext. In *Analyzing Microtext: Papers from the 2011 AAAI Workshop* (pp. 74–79). San Francisco, CA, USA: AAAI. Retrieved from http://www.aaai.org/ocs/index.php/WS/AAAIW11/paper/view/3987

Zhang, W., Yoshida, T., & Tang, X. (2008). TFIDF, LSI and multi-word in information retrieval and text categorization. In *2008 IEEE International Conference on Systems, Man and Cybernetics* (pp. 108–113).

Zhang, X., Fuehres, H., & Gloor, P. A. (2012). Predicting Asset Value through Twitter Buzz. In J. Altmann, U. Baumöl, & J. B. Krämer (Eds.), *Advances in Collective Intelligence 2011* (pp. 23–34). Berlin, Heidelberg: Springer Berlin Heidelberg.

Zhao, D., & Rosson, M. B. (2009). How and Why People Twitter: The Role That Micro-blogging Plays in Informal Communication at Work. In *Proceedings of the ACM 2009 International Conference on Supporting Group Work* (pp. 243–252). New York, NY, USA: ACM.

Zhu, C., Tang, J., Li, H., Ng, H. T., & Zhao, T. (2007). A Unified Tagging Approach to Text Normalization. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (pp. 688–695). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from

http://www.aclweb.org/anthology/P/P07/P07-1087

LIST OF PUBLICATIONS AND PAPER PRESENTED

Saloot, M. A., Idris, N., Aw, A., & Thorleuchter, D. (2014). Twitter corpus creation: The case of a Malay Chat-style-text Corpus (MCC). *Digital Scholarship in the Humanities*. Retrieved from http://dsh.oxfordjournals.org/content/early/2014/12/13/llc.fqu066.abstract

Saloot, M. A., Idris, N., & Mahmud, R. (2014). An architecture for Malay Tweet normalization. *Information Processing & Management*, *50*(5), 621–633.