A Study on Knowledge Acquisition from Social Media Data and Its Application

（ソーシャルメディアからの知識獲得とその応用に関する研究）

山梨大学大学院
医学工学総合教育部
博士課程学位論文

2019年3月

Apichai Chan-udom

Presented a Doctoral Viva Voce, Information and Mechanical System Engineering
(System Solution Engineering),
Interdisciplinary Graduate School of Medicine and Engineering,
University of Yamanashi, Japan, 2019
Date : February 6th, 2019
Venue : Division of Engineering,
Interdisciplinary Graduate School of Medicine and Engineering,
University of Yamanashi, Japan
Hosted by : University of Yamanashi, Japan

Acknowledgements

I would like to express my sincere gratitude to my advisor Professor Dr. Suzuki Yoshimi for the continuous support of my PhD study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. Without his guidance and constant help this PhD would not have been achievable.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Munehisa Tomoo, Prof. Fukumoto Fumiyo, Prof. Watanabe Yoshimichi, Prof. Nishizaki Hiromitsu and Prof. Tanzawa Tsutomu, for their encouragement, insightful comments, and valuable questions.

And I would like to thank Suzuki-ken (laboratory mates), for the presentations and discussions on seminars for the last four years. Also, I am grateful to the university officers for many supports related to my study.

Finally, I gratefully acknowledge the scholarship received towards my doctoral study from the Rajamangala University of Technology Srivijaya, Thailand.

Abstract

Twitter is a micro-blog service, people can tweet or re-tweet or follow, what is happening at any moment a time, anywhere in the world that is an effective way to spread out information and express opinions or their feelings. Nowadays, the massive volume of tweets has become an interesting source for studying in various aspects. Text from social media provides a set of challenges, informal language, spelling errors, abbreviations, unknown word, and special characters. In our study consist of 3 sections.

The first section, we studied for solving a problem of ambiguous words in the tweets, especially related to the technology domain. The issue is that tweets are short text, so the ambiguous words in tweets do not have enough context information to decide the right meaning. Therefore, we introduced a way to extend the context of the ambiguous word in a specific Twitter data (technology domain) by taking advantage of the word vectors (or words embedding) which was created by word2vec [Mikolov et al., 2013a]. The result shows that a sense of 30 target ambiguous words in the technology domain tweets attained at 83.33% accuracy.

The second section, we presented a method for detecting new-words in Twitter data which do not appear in the WordNet database and defining their sense utilize the abstract information of Wikipedia. To definition new word sense, our proposed method utilizes words embedding from word2vec to expand their context words and selecting the most meaningful sentence from the Wikipedia abstract information page of each new word.

The final section, we describe how to classify Thai language tweets. In this study, we collected experimental data from Twitter which have a specific hashtag. Even if tweets which have the same hashtags, there are many kinds of tweets; suggesting solutions, emotional tweets, news report, and other tweets. We used the tweets data of Thai cave rescue incident; the 12 boys and their coach soccer team trapped inside the cave after heavy rain and water flood (during June 23-July 12, 2018) in Thai language and Japanese for our experiment. Under the idea when someone would like to get inspiration and advice, he may get some advice from tweets which are described suggesting solutions. We conducted some experiments for tweet classification using five machine learning algorithm. The experimental results showed that we can obtain good results. Moreover, we compared the classifying results of tweets written in Thai and the results of tweets written in Japanese. The compared results have shown Japanese tweets classified results obtained accuracy higher than Thai tweets because of word-embedding quality (vocabulary size and corpus size) and Japanese tweets longer than Thai tweets on average.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Motivation and Background

Nowadays, there are numerous online documents such as newspaper web, micro-blogs, and web digest, etc., we do not have enough time for reading all of them. Twitter is a social micro-blogging service that supported more than 40 languages, approximately 500 million tweets a day based on worldwide users about 326 million.

In economics studies or social science, or even though the business field; a portion of the study must be understanding human behavior or consumer behavior. From online documents or social media data, how we get information? To forecast trends of customer behavior or economic trend, and to answers questions like such:

- What is most people's opinion about Brexit[1]?
- How did people feel about North Korea missile? And what are people's suggestions?
- What emerging technologies are going on?
- ...?

From the literature review, many researchers use Twitter data for studying in various aspects. [Kanungsukkasem and Leelanupab, 2014] have studied the relationship between Twitter data and financial instruments (stock), according to online social behavior in Twitter related to movement of the financial markets or any financial instrument, based on hypothesis refer to behavioral economics effect to behavioral finance of investors [Bollen et al., 2011].

Japan earthquake frequently occur, Japanese peoples tweet their feelings, opinions, and situation, and also mention disaster information. There are many researchers contributed

[1] British exit, is the impending withdrawal of the United Kingdom from the European Union.

their study using Twitter data related to earthquakes disaster in Japan. For instance, Robert Thomson et al. presented an idea of selecting a credible Twitter information source related to the Fukushima disaster. The idea considerate on the Twitter user profile, profile anonymity proved to be correlated with a higher propensity to share information from low credibility sources [Thomson et al., 2012]. Hiroko Wilensky has studied a qualitative investigation of the use of Twitter by the stranded commuters and their supporters in the Tokyo metropolitan area immediately after the earthquake [Wilensky, 2014]. Tatsuya Aoki et al. proposed detecting of severe mental distress concerning earthquakes utilize Twitter as a sensor [Aoki et al., 2017].

Currently, Thailand does not have information systems or application to keep taking peoples on disaster situation or epidemic situation such as water flood, typhoon, Influenza A(H1N1), Zika virus disease, and so on.

In natural language processing research area, based on the advantage of Twitter data which mention above, we decided to employ Twitter data for our studying and do research.

1.2 WordNet

Miller, Fellbaum, et al. developed WordNet[Fellbaum, 1998]; a lexical thesaurus that serves as a widely used resource in computational linguistics and natural language processing applications.

The WordNet lexical database; Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations.

In our study, we used WordNet 3.0 database files. The noun words in WordNet 3.0 consists of 117,798 words. These words are organized into fine-grained classification 82,115 synsets each consists of synonyms. The average number of words in a synset is very few, 1.24 per a synset.

1.3 Word2vec

Word2vec[Mikolov et al., 2013a] was proposed by a team of researchers led by Tomas Mikolov at Google. There are two main algorithms architectures in word2vec; continuous bag-of-words shows in Figure 1.1 and continuous skip-gram shows in Figure 1.2 as below. These models are shallow neural networks that are learn word vector representations (one-hot-vector) in contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, with each unique word in the corpus being assigned a corresponding vector in the space.

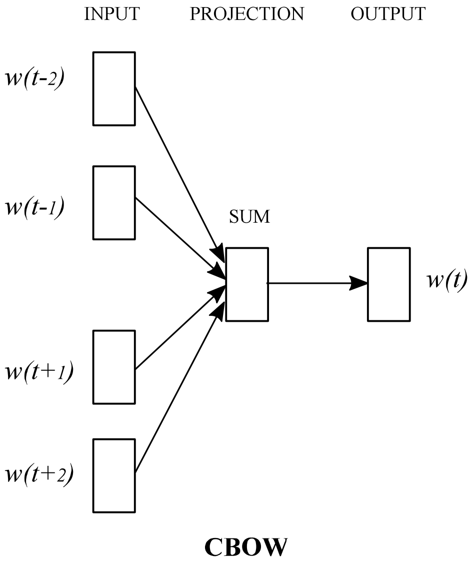


Fig. 1.1 The continuous bag-of-words architecture

The CBOW[2] Model architecture takes the context of each word as the input and **tries to predict the word corresponding to the given context**.

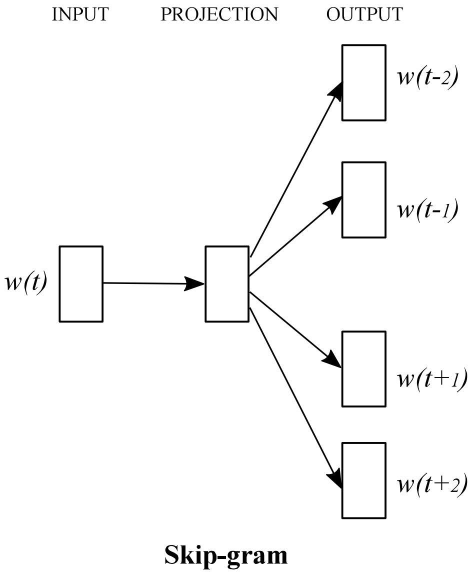


Fig. 1.2 The continuous Skip-gram architecture

The skip-gram[Mikolov et al., 2013b] architecture is similar to CBOW, but instead of predicting the current word based on the context, it predicts words within a certain range before and after the current word (**predicts the surrounding words given the target word**).

[2] continuous bag-of-words

The objective of the Skip-gram model is to maximize the average log probability, definition as below:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c,j\neq 0}\log p(w_{t+j}|w_t) \tag{1.1}$$

where c is the size of the training context (which can be a function of the center word w_t). The basic Skip-gram formulation defines $p(w_{t+j}|w_t)$ using the softmax function:

$$p(w_O|w_I) = \frac{exp(v_{w_O}^{'\ T} v_{w_I})}{\sum_{w=1}^{W} exp(v_w^{'\ T} v_{w_I})} \tag{1.2}$$

where v_w and $v_w^{'}$ are the "input" and "output" vector representations of w, and W is the number of words in the vocabulary.

1.3.1 Hyperparameters of Word2vec tool

To consider the hyperparameters of Word2vec tool we refer to the web page information about Word2vec [code., 2013], the main parameters of word2vec shows as below:

- architecture: skip-gram (slower, better for infrequent words) vs CBOW (fast),
- the training algorithm: hierarchical softmax (better for infrequent words) vs negative sampling (better for frequent words, better with low dimensional vectors),
- sub-sampling of frequent words: can improve both accuracy and speed for large data sets (useful values are in range 1e-3 to 1e-5),
- dimensionality of the word vectors: usually more is better, but not always,
- context (window) size: for skip-gram usually around 10, for CBOW around 5.

1.3.2 Comparison CBOW Architecture and Skip-gram Architecture on Word Similarity Measures

The Word2vec tool proposes a good idea for converting(encoding) context information of words to a vector representation which utilizes a shallow neural network that contains CBOW architecture and skip-gram architecture.

In word similarity measure using context information which based on hypothesis the words occur in a shared context they are similar. We would like known between CBOW and skip-gram which one is good for predicting words in a given context(shared context)?

To comparative between CBOW and skip-gram, we use Spearman's correlation coefficient and Pearson correlation coefficient between Cosine similarity measure pairs of word vector with human judgment benchmark dataset that shows in Table 1.1 below. The result shows that skip-gram architecture outperformed on word similarity measures and word relatedness measures.

Table 1.1 Spearman and Pearson correlation coefficient between human and Cosine similarity measure using word vectors from word2vec tool compare between skip-gram and CBOW

Benchmark Data	CBOW		Skip-gram	
	Spearman	Pearson	Spearman	Pearson
WS-353-SIM.	0.695	0.691	**0.749**	**0.741**
WS-353-REL.	0.650	0.621	**0.682**	**0.660**
RG-65	0.673	0.649	**0.716**	**0.687**

1.4 Definition

Definition of 'Technology':

Technology is the collection of techniques, skills, methods, and processes used in the production of goods or services or in the accomplishment of objectives, such as scientific investigation. Technology can be the knowledge of techniques, processes, and the like, or it can be embedded in machines to allow for operation without detailed knowledge of their workings.[Wikipedia, 2018]

New word:

In our study, the new word means that "word" does not exist in WordNet database files and that word must able to get abstract from Wikipedia.

We able to use word "neologism" instead of the word "New word", the definition of neologism from WordNet lexical is "a newly invented word or phrase."

Stop word:

Stop words mean these words frequently occur in documents that ineffective for distinct documents. For example, all, a, an, the, that, etc.

Chapter 2

Word Similarity

2.1 Word Similarity Measuring

From the literature review, there are two main similarity measures:

- (1) Thesaurus-based similarity measures,
- (2) Distributional similarity measures.

2.1.1 Thesaurus-based measures

Meng et al. proposed a review of word similarity measures based on WordNet [Meng et al., 2013] as below.

The Shortest Path based Measure:

Rada, et al. introduced the conceptual distance measure which is the length of the shortest path between two concepts (c_1 and c_2)[Rada et al., 1989]. Two concepts(synsets) c_1, c_2 are similar if they are near each other in the thesaurus hierarchy, the concept has 1 path to themselves.

$$sim_{path}(c_1, c_2) = \frac{1}{pathlen(c_1, c_2)} \tag{2.1}$$

$pathlen(c_1, c_2)$ = 1+ number of edges in the shortest path in the hypernym graph between synset nodes c_1 and c_2.

Leacock-Chodorow Measure:

The measure suggested by [Leacock and Chodorow, 1998] considers only the *is* a hierarchies of nouns in WordNet. To determine the semantic similarity of two synsets, the shortest path between the two in the taxonomy is determined and is scaled by the depth of the taxonomy. The following formula is used to compute semantic similarity:

$$sim_{lch} = -log\frac{shortestpath(c_1,c_2)}{2*D} \tag{2.2}$$

Where D is the maximum depth of the taxonomy.

Wu & palmer Measure:

Wu and Palmer extend this measure by incorporating the depth of the Least Common Subsummer (LCS) [Wu and Palmer, 1994]. The LCS is the most specific concept two concepts share as an ancestor. In this measure, the similarity is twice the depth of the two concepts LCS divided by the product of the depths of the individual concepts as defined below.

$$sim_{wup} = \frac{2*depth(lcs(c_1,c_2))}{depth(c_1)+depth(c_2)} \tag{2.3}$$

Resnik's Measure:

Philip Resink proposed the measure which using information content of a concept measures the specificity or the generality of that concept [Resnik, 1995].

$$P(c) = \frac{\sum_{w \in words(c)} count(w)}{N} \tag{2.4}$$

Information content: $IC(c) = -logP(c)$
Lowest common subsume between c_1 and c_2: $LCS(c_1,c_2)$

$$sim_{resnik}(c_1,c_2) = -logP(LCS(c_1,c_2)) \tag{2.5}$$

Lin's Measure:

The measure based on information content of concepts, was proposed by [Lin, 1998b].

$$sim_{Lin}(c_1,c_2) = \frac{2logP(LCS(c_1,c_2)}{logP(c_1)+logP(c_2)} \tag{2.6}$$

Jiang-Conrath Measure:

The measure was introduced by [Jiang and Conrath, 1997] addresses the the limitations of the Resnik measure.

$$sim_{jiang}(c_1,c_2) = \frac{1}{logP(c_1) + logP(c_2) - 2logP(LCS(c_1,c_2))} \quad (2.7)$$

Adapted Lesk (Extended Gloss Overlaps):

The Overlaps measure, two concepts are similar if their glosses contain similar words. To adaptive, S. Banerjee and T. Pedersen proposed the extended gloss overlaps measure (adapted Lesk algorithm) [Banerjee and T., 2003] that based on the number of shared words (overlaps) in their definitions (WordNet glosses), and expands the glosses of the words being compared to included glosses of concepts that are directly related according to hierarchical in WordNet.

$$sim_{eLesk}(c_1,c_2) = \sum_{r,q \in RELS} overlap(gloss(r(c_1)), gloss(q(c_2))) \quad (2.8)$$

2.2 Vector similarity measures

In our study, we utilize vector similarity measures for word similarity; two words are similar if their vectors are short-distance or closely. The vector similarity measures are consist of Manhattan distance, Euclidean distance, Cosine similarity, Jaccard similarity coefficient, Dice measure, Kullback-Leibler divergence, Jensen-Shannon divergence, and Hellinger distance their definition shows as below.

1. Manhattan distance:

$$dist_{Manhattan}(\vec{x},\vec{y}) = \sum_{i=1}^{n} |\vec{x}_i - \vec{y}_i| \quad (2.9)$$

2. Euclidean distance:

$$dist_{Euclidean}(\vec{x},\vec{y}) = \sqrt{\sum_{i=1}^{n} (\vec{x}_i - \vec{y}_i)^2} \quad (2.10)$$

3. Cosine similarity measure:

$$sim_{cosine}(\vec{x},\vec{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2)}\sqrt{(\sum_{i=1}^{n} y_i^2)}} \quad (2.11)$$

4. Jaccard's coefficient measure:

$$sim_{Jaccard}(\vec{x},\vec{y}) = \frac{\sum_{i=1}^{n} min(x_i,y_i)}{\sum_{i=1}^{n} max(x_i,y_i)} \tag{2.12}$$

5. Dice measure:

$$sim_{Dice}(\vec{x},\vec{y}) = \frac{2 * \sum_{i=1}^{n} min(x_i,y_i)}{\sum_{i=1}^{n} max(x_i,y_i)} \tag{2.13}$$

6. Kullback-Leibler divergence:

$$D_{KL}(P||Q) = \sum_{i=1}^{n} P(x_i) * \log(\frac{P(x_i)}{Q(y_i)}) \tag{2.14}$$

where P and Q are probability distributions that defined on the same probability space and $Q(y_i) \neq 0$.

7. Jensen-Shannon divergence:

$$sim_{JS} = D_{KL}(\vec{x}||\frac{\vec{x}+\vec{y}}{2}) + D_{KL}(\vec{y}||\frac{\vec{x}+\vec{y}}{2}) \tag{2.15}$$

8. Hellinger distance measure:

$$sim_{Hellinnger}(\vec{x},\vec{y}) = \frac{\sum_{i=1}^{n} \sqrt{x_i y_i}}{(\sum_{i}^{n} x_i)(\sum_{i}^{n} y_i)} \tag{2.16}$$

an each vector must be normalized to 1 in L_1 norm: $\sum_{i=1}^{n} x_i = 1$ and $\sum_{i=1}^{n} y_i = 1$.

In our experiments, we used three vector similarity measures for comparative these are cosine similarity, Jaccard's coefficient, and Hellinger distance. Notation, $\vec{x},\vec{y}$: a vector of each word gloss, where n is a vector size (number of elements in a row matrix)

2.3 Our Proposed Approach

From literature review, we found concepts of word similarity measuring can be divided into two main ideas are:

a) The word similarity measure based on the dictionary or using relationship structure of words in the thesaurus. There were collected and presented by Ted Pedersen [Pedersen et al., 2004], and Lingling Meng [Meng et al., 2013]. The word similarity measurements based on dictionary can be measured at the level of word meaning(sense) of each word. However, the performance of the measurement depends on the quality of the relationship structure of words in the dictionary or the thesaurus only, in some case the hierarchy relatedness structure of word senses in WordNet (or dictionaries) not all perfect. Unfortunately, the context information which is important does not use for word similarity measure under this concepts.

b) Another is the measuring the word similarity using context information. Under this concept, the words occur in same contexts frequently we can point that words are similar such as “I read a book” and “I read a magazine” the word “book” and “magazine” occurred in same contexts, that means “book” and “magazine” are similar. Many researchers have published research paper about utilize context information such as D. Hindle [Hindle, 1990], Tomas Mikolov et al [Mikolov et al., 2013a], etc.

The important factors effect to the performance of this concept is quantity and quality of text corpus and the computer methodology that can be processed the context information properly such as machine learning, neural network, etc. Presently, the software library is used for constructing a word vector representation using context information e.g., Word2vec, Glove, and FastText.

The Word2vec creates a vector of a word consists of context features. We use the vector representation of words from Word2vec to find a set of similar words of target words, that we called the "related words", these are the words that frequently appear in same contexts as the target word. it may be difficult to identify which word of them is most similar to the target word.

For example, “we drink tea”, “we drink coffee”, “we drink beer”, “we drink wine”, the words “tea”, “coffee”, “beer”, and “wine” which occurred in same contexts. If a number of occurring times of each word are equally, what is the word that most similar to the word “tea”? For depth measuring on word similarity used context information only that not enough, we should consider a definition of each word also. Therefore, we have to use the word similarity measure based on the dictionary or thesaurus as the second stage repeatedly.

We proposed the new approach for the word similarity measure that provides better efficiency. It combines two key concepts for measuring the word similarity together. In addition, we apply the new word similarity approach for word sense disambiguation on a specific domain, that is the technology domain based on tweet corpus. The two-stages word similarity measure as shown in Figure 2.1.

In the two-stages similarity measure, to compute the similarity score between an each gloss vector of the target word and any gloss vector of its related words is defined as follows:

$$sim(tws_i, rws_j) = sim_{s2}(tws_i, rws_j) \times (1 + w2v(tw, rw)) \tag{2.17}$$

where $sim(tws_i, rws_j)$ is the similarity value between any gloss vector of target word (tws_i) and any gloss vector of its related word (rws_j), and $w2v(tw, rw)$ is the cosine similarity value between the target word (tw) and an each its related word (rw) utilize word vector which obtained from Word2vec, and
$sim_{s2}(tws_i, rws_j)$ is the similarity value in the second-stage between a gloss vector of target word and a gloss vector of its related word.

For selecting the most similar word of the target word, our system chooses the word from the highest score of the pairs. To identifying the best sense of target word, our system summarizes the similarity score of each sense of target word and point to which sense has a maximum summation score is the best sense of target word.

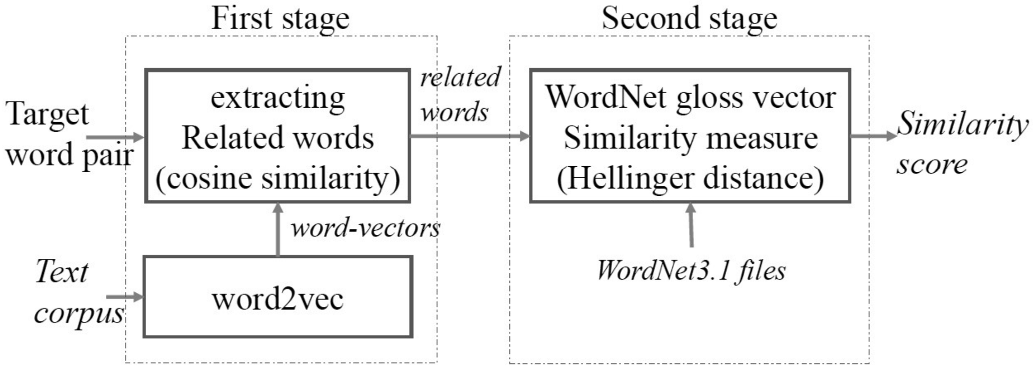


Fig. 2.1 The two-stages word similarity measure.

We use Cosine similarity measure a word vector pairs of benchmark dataset for copare the word vector source that shows in Table 2.1 below.

Table 2.1 Comparative the word vector sources using Cosine similarity measure

Benchmark Data	MC30		RG65		WS-353-SIM.	
	ρ	r_s	ρ	r_s	ρ	r_s
word2vec (cosine measure)	0.66	0.65	0.63	0.62	0.67	0.68
WordNet gloss vector (Hellinger)	0.70	0.70	0.74	0.72	0.55	0.60
Proposed method	**0.86**	**0.78**	**0.82**	**0.77**	**0.72**	**0.73**

Where ρ mean Spearman, r_s mean Pearson.

2.4 Customization Hyper Parameters of Word2vec

The data sets(tweets data) in the experiment is not large, and the data quality is not well corpus because tweets often include some symbols such as emotional(Emoji/emoticon) symbols e.g.,☺, ☹, ♡. Moreover, an each tweet is limited 140 characters, and its writing style is the uncommonly written format. Therefore, it is necessary to customize parameters of Word2vec for training with tweets data.

For testing to customize Word2vec training parameters, we consider in cosine similarity value between the word “*wireless*" and “*bluetooth*" to compare the effect of Word2vec train parameter.

In the first round of testing, we varied a vector size, *i.e.*, the vector size is step up in 10 from 40 to 220 and compare two training algorithms, hierarchical softmax and negative sampling. As a result, we found that when the vector size is small, cosine similarity value provided the highest value. Moreover, we found that hierarchical softmax training algorithm had a better performance than negative sampling which is shown in Figure 2.2.

In the second round of testing, we varied the window size parameters. It is searched in steps of 1 from 4 to 10 which is illustrated in Figure 2.3. We can see from Figure 2.3. As a result, the obtained cosine similarity value has become to increase when the window size is small.

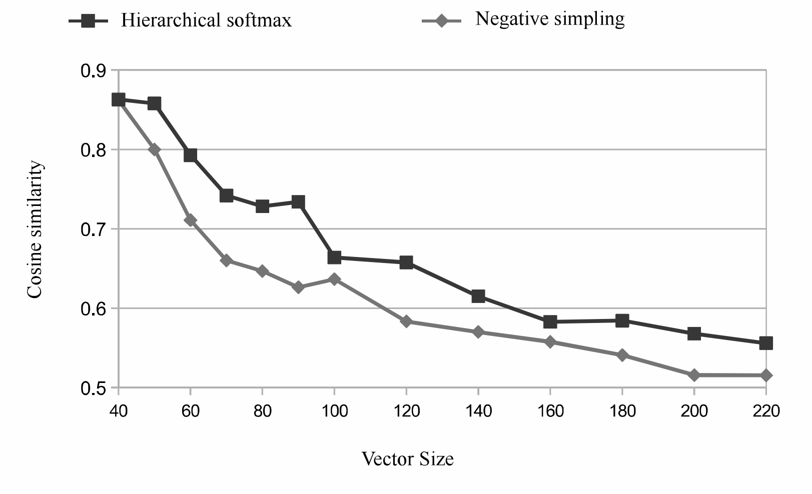


Fig. 2.2 Comparison between hierarchical softmax and negative sampling training algorithm in Word2vec

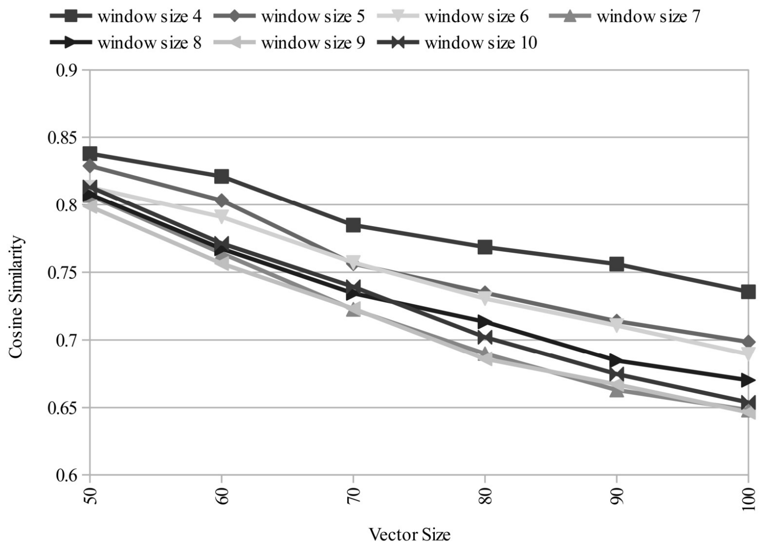


Fig. 2.3 Cosine similarity for varying window size and vector size in Word2vec

From our experiments result, we found that CBOW model has good performance than skip-gram model. And finally, we can design our training command for train Word2vec with tweets data as below:

```
    ./word2vec -train tweets.txt -output vectorFile -cbow 1 -window 4
-size 80 -negative 0 -hs 1 -sample 1e-5 -threads 20 -binary 1 -iter 25
-min-count 6
```

2.5 Identification of Word Sense in Twitter Data based on WordNet Glosses

With the exponential growth of information on the Internet, the short text such as search snippets and Twitter data are widely available on the Internet. There are some semantic-oriented application for these data need to recognize word senses to detect which words may be similar to each other. To alleviate the sparseness of a sentence extracted from Twitter data, we focus on expanding a short sentence with knowledge extracted from the auxiliary tweet corpus. To do this, we applied Word2vec to the tweet corpus and constructing related words of a target word. We identified a sense of the target word by calculating a similarity between two vectors represented, one is a vector of an each gloss(definition of each index word in dictionary) of the target word, and the other is an each gloss of its related words which are obtained by the Word2vec. The result shows that a sense of 30 target words from the technology domain tweet data attained at 83.33% accuracy.

2.5.1 Introduction of Word Sense Identification

Many semantic-oriented application such as Opinion Mining and Question Answering need to recognize which words may be similar to each other. The earliest known approach for word sense identification used corpus-based statistics [Hindle, 1990; Lin, 1998a]. The similarity measures based on distributional hypothesis compared a pair of weighted feature vectors that characterize two words. There have been numerous methods that attempt to calculate semantic similarity [Kusner et al., 2015; Mehwish and Muhanmmad, 2010; Salton and Buckley, 1988]. Other approaches in sense identification utilized fine-grained and large-scale semantic knowledge like WordNet, COMLEX, EDR dictionary, and Bunrui-Goi-Hyo [Meng et al., 2013; Rodriguez and Egenhofer, 2003]. A well-known technique in dictionary-based sense identification was the calculating semantic similarity between a context (sentence) of the target word and a gloss representing a sense of the target word in the semantic knowledge.

However, unlike textual corpora such as newspapers and scientific papers, Twitter data often consists of short length of text. The following examples show two sentences extracted from BBC news and tweet data including the target word, “drone”. The word “drone” has (at least) five noun senses in the WordNet3.1 including radio-controlled aircraft. We can see from the example 1 that the drone is a radio-controlled aircraft as the word occurs ‘Unmanned Aerial Vehicles’, and ‘Remotely Piloted Aerial Systems’. However, it is difficult to identify a sense of drone in the example 2 because it occurs only a few common words.

Example 1. BBC news

> Drones: What are they and how do they work?
> (http://www.bbc.com/news/world-south-asia-10713898)
> “... To the military, they are UAVs (Unmanned Aerial Vehicles) or RPAS (Remotely Piloted Aerial Systems). However, they are more commonly known as drones. ...”

Example 2. Tweet data

> “I was on podcast with a great group of tweeps talking tech acquisitions IoT drones”

In this study, we propose a method for identifying word senses in Twitter data. To alleviate the sparseness of a sentence extracted from Twitter data, we expand a short sentence with knowledge extracted from auxiliary tweet corpus. We applied the Word2vec to the tweet corpus and represented the target word as a vector and use its vector to find its similar words(related words), 40 words. We identified a sense of the target word by calculating a similarity between two vectors, one is a vector of an each glossary of the target word, and another one is a glossary of its related words. The vectors represented a glossary of words that we called “gloss vector”. Our experiments used WordNet 3.1 sense inventory.

2.5.2 System Outline

Our system consists of five (main) processes as shown in Figure 2.4. The first process is Tweets Acquisition, tweets acquisition program was developed with C# language and using Tweetinvi C# library, this process collected tweets through Twitter API service. The second process is POS Tagging using the TreeTagger(Windows version) [Schmid, 1994]. The third process is Stemming, this process replaced a word with its lemma(or base form). The fourth process determined similar words using word-vectors from Word2vec which result is a set of related words(40 words) of the target word.

The last process is Sense Identification, that used our proposed method (two-stages word similarity measure), in this process to find a similar meaning between sense pairs of target words and its related words based on WordNet3.1 Glosses that was represented as a gloss vector of each sense – using Cosine similarity measurement and Jaccard coefficient and Hellinger distance [Dingding et al., 2015] for comparison.

Finally, for identifying the best sense of target word, our system summarizes the similarity score of each sense of target word and decide the best sense of target word according to the domain. In our experiments, domain is technology because our tweet corpus was collected by technology keywords.

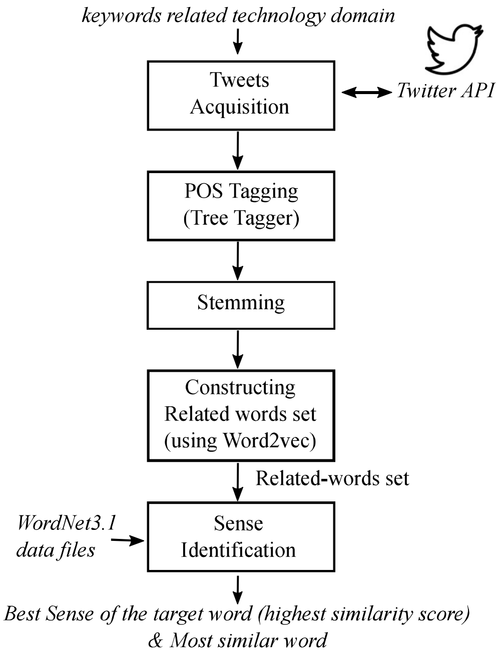

keywords related technology domain
Tweets Acquisition
Twitter API
POS Tagging (Tree Tagger)
Stemming
Constructing Related words set (using Word2vec)
Related-words set
WordNet3.1 data files
Sense Identification
Best Sense of the target word (highest similarity score) & Most similar word

Fig. 2.4 System outline

2.5.3 Experimental Results

To comparative similarity measurements, we examined the effect of each similarity measurement utilizes the gloss vector for computing similarity score between sense pairs. The vector represented a glossary(definition) of a word, that glossary obtained from WordNet 3.1 data files. To evaluate similarity measurement using 30 testing words(mentioned above) which were assigned the best sense according to technology context. We did experiments with 20 data sets(tweets data) for every similarity measure methods and compared by accuracy percentage on average as shown in Table 2.2. As the result, the best performance was obtained from the Hellinger distance with TF-IDF weighting. Moreover, we found that removing stop words from a gloss vector that can improve overall performance.

Table 2.2 Similarity improvement for six methods

Methods	without stop words	with stop words	Improvement rate
Hellinger Distance (TF-IDF)	71.04	48.02	23.02%
Cosine (Binary)	69.97	43.30	26.67%
Cosine (TF-IDF)	69.55	51.98	17.57%
Hellinger Distance (term freq.)	69.44	38.89	30.55%
Jaccard Coefficient	69.31	38.33	30.98%
Cosine(term freq.)	68.44	37.29	31.15%

For customizing a gloss vector, we examined gloss vectors consist of (1) synset-words (2) hypernym gloss, and (3) hypernym synset-words. Table 2.3 shows the results.

In Table 2.3, Gl means the method using Glosses of WordNet, RS indicates without stop words, AS refers to Adding synset-words, AHG means Adding Hypernym Glosses and AHS indicates Adding Hypernym synset-words. In each column, "√" and "-" indicate with and without each information. We obtained the highest accuracy with the gloss vector added synset-words and hypernym synset-words.

Table 2.3 Accuracy for a gloss vector customization

Glossary Customization					Accuracy
Gl	RS	AS	AHS	AHG	rate
√	√	√	√	-	72.87
√	√	√	√	√	71.30
√	√	-	√	-	70.14
√	√	√	-	√	69.58
√	√	-	√	√	68.98
√	√	√	-	-	68.52
√	√	-	-	√	68.06
√	√	-	-	-	67.55
√	-	-	-	-	40.28

2.5.4 Discussion

The method attained at 83.33% accuracy in word sense identification. This shows the effectiveness of the method.

In the failure case, the word “cookie” was not identified to “a short line of text that a website puts on your computer’s hard drive when you access the website”, but become “any of various small flat sweet cakes”, while our tweets data was collected by technology keywords. In the tweets data, we found that “cookie” often co-occurred with words about kind of “desserts” or “foods” e.g., *cooking machine*, jam, chocolate, and so on.

2.5.5 Conclusion

In this paper, we proposed a method for identifying word senses in Twitter data. The result showed the effectiveness of the method. For future work, we will conduct additional experiments using other new words in Twitter data.

2.6 Related Work

From the literature review, Banerjee and Pedersen proposed the extended gloss overlaps measure (adapted Leskalgorithm)[Banerjee and T., 2003]; based on the number of shared words (overlaps) in their definitions (WordNet glosses), expands the glosses of the words being compared to included glosses of concepts that are directly related according to hierarchical in WordNet.

Patwardhan and Pedersen proposed the gloss vector measure [Patwardhan and Pedersen, 2006]; A gloss vector of each concept(synset) builds from second order co-occurrence that derive from first order co-occurrences matrix. Using cosine for determine similarity score between gloss vectors. The disadvantage of this method is vectors have a very large number of dimensions (sparse vector).

Chapter 3

Detecting New Words in Twitter Data

3.1 Introduction to New Words in Twitter data

Many semantic-oriented applications such as Question Answering, Text Categorization, Sentiment Analysis, and Text Summarization by using Twitter data need to recognize which words may substitute each other in a meaning consistency ([Lin and Pantel, 2001],[Barzilay and McKeown, 2001],[Quirk et al., 2004] and [Metzler et al., 2011]).

The WordNet 3.1 consists of noun words 117,798 words. These words are organized into fine-grained classification, 82,115 synsets, each consists of synonyms. The average number of words in a synset is very few, 1.24 per a synset. The current version of WordNet database files released in December 2006 that is more than 10 years ago. Since new words are constantly being created, especially the technical term related to emerging technology, new meaning is also being assigned to the existing words, to update WordNet database files are quite difficult and expensive. As a result, the performance of semantic-oriented application utilizes WordNet is decreased.

Therefore, considering this resource scarcity problem, semantic tagging of words which do not appear in the WordNet resource but appear in Twitter data has been an interest since the earliest days when a number of large scale corpora have become available.

Twitter data has been an interest since the earliest days when a number of large-scale corpora have become available and also in Twitter corpus easier to found new-words then another corpus.

We proposed a method for detecting new-words in Twitter data which do not appear in the WordNet resource and defining their glossary using abstract information from Wikipedia. The Wikipedia represents a rich semantic network connecting entities and concepts, enabling it as a valuable source for knowledge gathering.

From our experimental Twitter data have been found the new words (the new technical terms in technology domain), a portion of new-words show as the Table 3.1, and the instance of tweets which a new word being shows in the Example 1 as below.

Table 3.1 Portion of new-words

new-word	frequency	new-word	frequency
smartphone	98,097	runtime	423
bluetooth	62,795	hotline	397
touchpad	21,852	netbook	345
ransomware	2,070	audiobook	172
widescreen	599	unicode	101
gis	555	adware	93

Example 1. An instance of a tweet

> "I was on podcast with a great group of tweeps talking tech acquisitions **IoT** drones"

Issue:

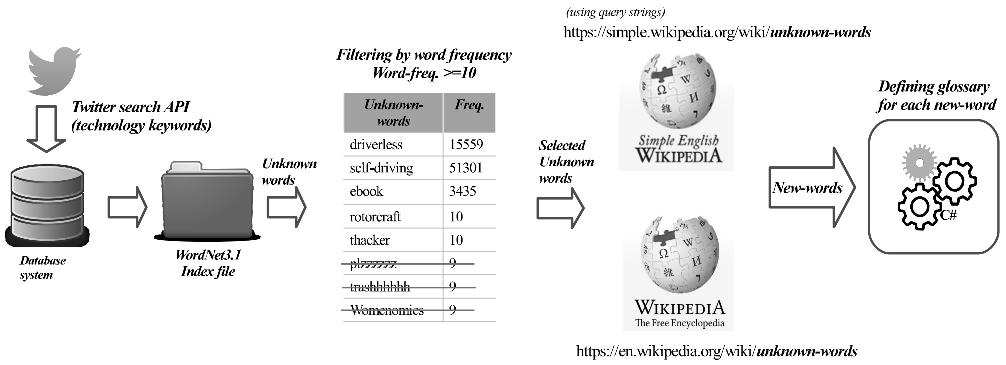
The word "IoT[1]" is an unknown-word, and it may be a new-word that appears in tweets with very few context words, that not enough information to guess the meaning of this unknown-word. Since new words have continuously occurred, nearly impossible to cover large and fast-changing of linguistic. We proposed ways to define a glossary of these new words which appeared in Twitter data related to the technology domain.

3.2 Detecting and Definition Framework

In Figure 3.1, our system collects Twitter data via Twitter search API using the technology keywords. Before saving to the database, our system program has been removing duplicates tweets, cleaning noise, i.e., emoji, user, RT, and URL. And after that, feed the Twitter data to Tree tagger for getting token lemma(their word stem) and utilize their word stem in the stemming process. We employ the WordNet index file to account unknown word.Later, filters unknown words with its frequency, if its frequency is less than 10 times, it will be taken out of unknown-words-list. And then puts a list of unknown words to the program for retrieving abstract information of each unknown word from Wikipedia, If that unknown word does not have Wikipedia abstract page it was skipped.

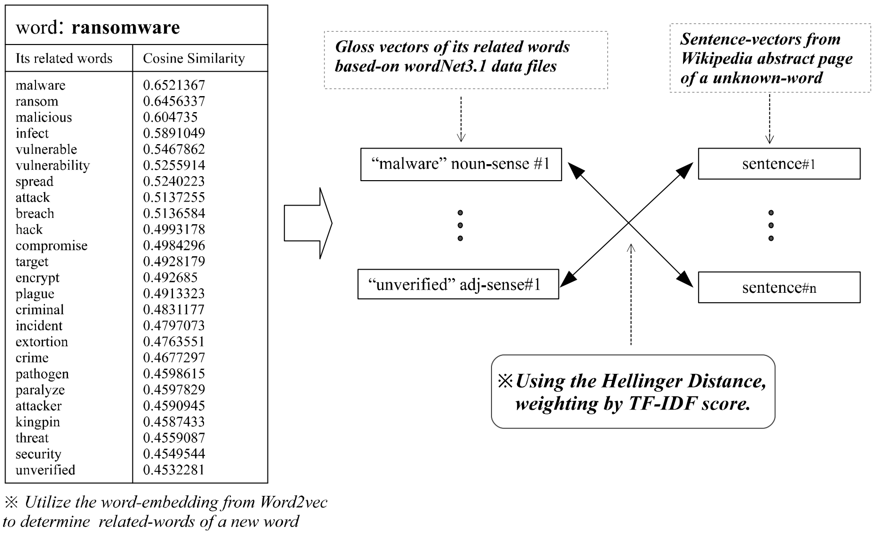
[1] https://en.wikipedia.org/wiki/Internet_of_things

In the process of glossary definition of each new word, our system gets abstract information of each new word from Wikipedia page via the query string method, after that break out an abstract text to the sentences, and represent a sentence to a co-occurred vector that weighting by TF-IDF score and a glossary vector of the related words are co-occurred vector weighting by TF-IDF also. The glossary inventory of the related words is WordNet3.1 database. For understanding easily, we demonstrate the definition process of the new word "ransomware" as below in Figure 3.2. The defined sense of the new word "ransomware" is the sentence which the highest summarized score shows in Table 3.2. We employ the Hellinger distance to across measure a similarity score between a gloss vector and a sentence vector.



Unknown-words	*Freq.*
driverless	15559
self-driving	51301
ebook	3435
rotorcraft	10
thacker	10
~~plzzzzzz~~	~~9~~
~~trashhhhhh~~	~~9~~
~~Womenomies~~	~~9~~

Fig. 3.1 Framework of New-words Detection and Definition Its Sense



word: **ransomware**

Its related words	Cosine Similarity
malware	0.6521367
ransom	0.6456337
malicious	0.604735
infect	0.5891049
vulnerable	0.5467862
vulnerability	0.5255914
spread	0.5240223
attack	0.5137255
breach	0.5136584
hack	0.4993178
compromise	0.4984296
target	0.4928179
encrypt	0.492685
plague	0.4913323
criminal	0.4831177
incident	0.4797073
extortion	0.4763551
crime	0.4677297
pathogen	0.4598615
paralyze	0.4597829
attacker	0.4590945
kingpin	0.4587433
threat	0.4559087
security	0.4549544
unverified	0.4532281

※ *Utilize the word-embedding from Word2vec to determine related-words of a new word*

Fig. 3.2 Demonstration of defining sense for a new word (ransomware)

The idea of our proposed method is the expanding context word of an unknown word (its related-words) which determine by cosine similarity measure between an unknown word vector and another word vectors. The word vector representation was generated by Word2vec using Twitter corpus. To defining a new word, we utilize glosses of its related words to selecting the closest similarity sentence in the abstract from its Wikipedia page.

Table 3.2 The summarized score of each sentence-vector

Sentence of abstract Wikipedia page of ransomware	**summation score**
It restricts access to the computer system that it infects, and demands a ransom be paid to the creator(s) of the malware.	**1.853904**
In June 2013 McAfee said it had collected over 250,000 unique samples of ransomware in the first three months of 2013.	1.018616
Others may simply lock the system and display messages intended to persuade the user to pay.	0.6571671
Some forms of ransomware encrypt files on the system's hard disk.	0.647037
CryptoLocker, a ransomware worm that surfaced in late-2013, had *collected an estimated $3 million USD before it was taken down by authorities.*	0.4315373
This is more than double the number of the previous year.	0.3837911
This is in order for the restriction to be removed.	0.1615445
Ransomware is a type of malware.	0.09256615

3.3 Experimental Setting

Our experiment of this section consists of 2 experiments; the first experiment we do experimental compare 2 abstract Wikipedia page source, en-wiki, and simple-wiki page. The second experiment we do experimental compare between our proposed method and other 2 ideas.

3.3.1 Testing-words Setup

In this experiment we choose the testing-words amount 66 words with Under mandatory conditions as below:

- They have only one sense(synset) in WordNet3.1 index,
- they have abstract page on the both: "https://en.wikipedia.org/wiki/" and "https://simple.wikipedia.org/wiki/".

The testing-word list used to evaluating our experimental process of defining a glossary are shown in Table 3.3 below.

Table 3.3 List of 66 testing-words

aircraft	electrician	html	malware	photovoltaic	subwoofer
amplifier	entrepreneur	igloo	manufacturing	porn	sunlight
axon	ethernet	internet	mathematics	pulsar	surveillance
bacteria	fedora	inventor	microphone	random	toddler
bodybuilding	fishery	ipod	multimedia	recruitment	transistor
burglary	gps	knowledge	myth	scientist	tray
chimpanzee	hdtv	laptop	nanotechnology	semantic	warranty
clover	headband	laser	newbie	sensor	wifi
creationism	headphone	librarian	ovum	shaman	workstation
desk	healthcare	limousine	petal	software	wrist
diploma	helicopter	linux	phobia	stochastic	wristwatch

3.3.2 ROUGE Evaluation

ROUGE measures recall: how much the words (or n-grams) in the human reference summaries appeared in the machine generated summaries [Chin-Yew, 2004].

$$Recall = \frac{number\ of\ overlapping\ words}{total\ words\ of\ reference\ summary} \tag{3.1}$$

$$Precision = \frac{number\ of\ overlapping\ words}{total\ words\ of\ system\ summary} \tag{3.2}$$

In our experiment, we decide to use ROUGE for evaluating the output of our system (the closest sentence/the sentence for defining a new word). The reference summaries are mean human-produced; in our experiment refer to WordNet glossary of a testing-word, WordNet glossary was defined by humans. The system summary is mean machine produced; in our experiment refer to the closest sentence as the output of our system; used to define a new word.

3.4 Experimental Results

The first experimental for comparison Wikipedia abstract page source they are two abstract pages available on Wikipedia, one is the en-wiki page and another one is the simple wiki page. We would like to know which abstract page source is more effective.

Table 3.4 Portion of results utilize abstract information from en_wiki page

Testing-words	en_wiki[2]page			
	Sentence No.	ROUGE(recall)	Precision	F1
aircraft	(1)	**0.333**	0.143	0.2
amplifier	(1)	**0.125**	0.071	0.090
wristwatch	(3)	**0.75**	0.3	0.429

Table 3.5 Portion of results utilize abstract information from simple_wiki page

Testing-words	simple_wiki[3]page			
	Sentence No.	ROUGE(recall)	Precision	F1
aircraft	(3)	0	0	0
amplifier	(13)	0.125	0.125	0.125
wristwatch	(1)	0.5	0.333	0.4

The results of this comparison experiment show as Table 3.4 and Table 3.5, from the results we got answer of our question as above that is the abstract from the en-wiki page source obtained higher effective than simple-wiki page source the reason is the abstract texts

[2] https://en.wikipedia.org/wiki/

from en-wiki page are more longer than abstract texts from simple-wiki page. The detail of a portion of testing words that show in Table 3.4 and Table 3.5 have illustrated as below.

WordNet glossary of 3 testing-words:

aircraft: a vehicle that can fly

amplifier: electronic equipment that increases strength of signals passing through it

wristwatch: a watch that is worn strapped to the wrist

The closest sentence; ID and sentence from en-wiki abstract page

(1) aircraft: An aircraft is a machine that is able to fly by gaining support from the air.

(1) amplifier: An amplifier, electronic amplifier or (informally) amp is an electronic device that can increase the power of a signal (a time-varying voltage or current).

(3) wristwatch: A wristwatch is designed to be worn around the wrist, attached by a watch strap or other type of bracelet.

The closest sentence; ID and sentence from simple-wiki abstract page

(3) aircraft: Some aircraft keep in the sky by moving air over their wings.

(13) amplifier: Electronic amplifiers have to be connected to electrical current or a battery to work.

(1) wristwatch: A watch is a small clock carried or worn by a person.

The second experiment is a comparison between our proposed method and 2 other ideas they are the selecting first-sentence on each abstract page and selecting the first-detected sentence which has a testing-word followed by the verb to be.

Table 3.6 Comparison result of three approaches using ROUGE(recall) for evaluation

	en-wiki page			simple-wiki page		
	avg(R)	avg(P)	avg(F1)	avg(R)	avg(P)	avg(F1)
Proposed method	0.208	0.111	0.134	0.192	0.164	0.159
First-sentence of each abstract page	**0.259**	**0.139**	**0.168**	0.214	0.176	0.177
First-detected sentence which has a testing-word followed by verb to be	0.237	0.129	0.155	0.209	0.181	0.175

[3] https://simple.wikipedia.org/wiki/

The results in Table 3.6 shows that selecting the first-sentence of each abstract page for defining a glossary(definition) of new-words is effective than another idea based on ROUGE evaluating. In a further study, we are looking the way improve our system performance by improving sentence similarity measure like as the WMD (word mover's distance) measure.

3.4.1 Related Works

Takale and Nandgaonkar published the measuring semantic similarity between words using web documents which used snippets from Wikipedia to overcome the new words issue [Takaleand and Nandgaonkar, 2010]. Huang et al. proposed the new word detection for sentiment analysis, the approach used for detecting new compound word in Chinese utilize lexical pattern distribution [Huang et al., 2014].

Chapter 4

Classifying Short Text in Social Media for Extracting Valuable Ideas

4.1 Introduction to Classifier

4.1.1 Naive Bayes

Naive Bayes methods [scikit learn, 2018a] are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Bayes' theorem states the following relationship, given class variable y and dependent feature vector x_1 through x_n :

$$P(y|x_1,\dots,x_n) = \frac{P(y)P(x_i,\dots,x_n|y)}{P(x_i,\dots,x_n)} \tag{4.1}$$

Using the naive conditional independence assumption that

$$P(x_i|y,x_1\dots,x_{i-1},x_{i+1},\dots,x_n) = P(x_i|y), \tag{4.2}$$

for all i this relationship is simplified to

$$P(y|x_1,\dots,x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1,\dots,x_n)} \tag{4.3}$$

Since $P(x_1,\dots,x_n)$ is constant given the input, we can use the following classification rule:

$$P(y|x_1,\dots,x_n)P(y) \propto \prod_{i=1}^{n} P(x_i|y) \tag{4.4}$$

$$\hat{y} = \underset{y}{\text{argmax}} P(y) \prod_{i=1}^{n} P(x_i|y), \tag{4.5}$$

and we can use Maximum A Posteriori (MAP) estimation to estimate $P(y)$ and $P(x_i, y)$; the former is then the relative frequency of class in the training set.

The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of $P(x_i, y)$.

In spite of their apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many real-world situations, famously document classification and spam filtering. They require a small amount of training data to estimate the necessary parameters. (For theoretical reasons why naive Bayes works well, and on which types of data it does, see the references below.)

Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality.

4.1.2 Gaussian Naive Bayes

GaussianNB implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}) \tag{4.6}$$

The parameters and are estimated using maximum likelihood.

4.1.3 Bernoulli Naive Bayes

BernoulliNB implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions; i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, boolean) variable. Therefore, this class requires samples to be represented as binary-valued feature vectors; if handed any other kind of data, a BernoulliNB instance may binarize its input (depending on the binarize parameter).

The decision rule for Bernoulli naive Bayes is based on

$$P(x_i|y) = P(i|y)x_i + (1 - P(i|y))(1 - x_i) \tag{4.7}$$

which differs from multinomial NB's rule in that it explicitly penalizes the non-occurrence of a feature that is an indicator for class , where the multinomial variant would simply ignore a non-occurring feature.

In the case of text classification, word occurrence vectors (rather than word count vectors) may be used to train and use this classifier. BernoulliNB might perform better on some datasets, especially those with shorter documents. It is advisable to evaluate both models, if time permits.

4.1.4 Support Vector Machines

Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression and outliers detection.

The advantages of support vector machines are:

- Effective in high dimensional spaces.
- Still effective in cases where number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

The disadvantages of support vector machines include:

- If the number of features is much greater than the number of samples, avoid over-fitting in choosing Kernel functions and regularization term is crucial.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation (see Scores and probabilities, below).

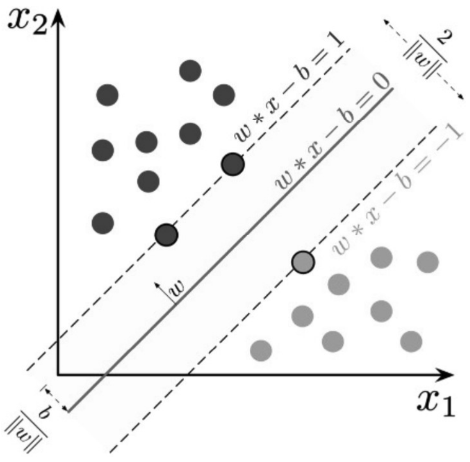


Fig. 4.1 support vector

The machine learn from the trained data point to maximize hyper-plane between 2 group of dataset.

4.1.5 Stochastic Gradient Descent - SGD classifier

Stochastic Gradient Descent (SGD) [scikit learn, 2018c] is a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions such as (linear) Support Vector Machines and Logistic Regression. Even though SGD has been around in the machine learning community for a long time, it has received a considerable amount of attention just recently in the context of large-scale learning.

SGD has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and natural language processing. Given that the data is sparse, the classifiers in this module easily scale to problems with more than 10^5 training examples and more than 10^5 features.

The advantages of Stochastic Gradient Descent are:

- Efficiency.
- Ease of implementation (lots of opportunities for code tuning).

The disadvantages of Stochastic Gradient Descent include:

- SGD requires a number of hyperparameters such as the regularization parameter and the number of iterations.
- SGD is sensitive to feature scaling.

4.1.6 Passive Aggressive Classifier

The passive-aggressive algorithms are a family of algorithms for large-scale learning. They are similar to the perceptron in that they do not require a learning rate. However, contrary to the perceptron, they include a regularization parameter C.[scikit learn, 2018b]

4.2 Classification of Thai Tweets Incident

The classification of Thai language tweets. In this study, we collected experimental data from Twitter data which have a specific hashtag. Even if tweet data which have the same hashtag, there are many kinds of tweets: news report, emotional tweets, suggestions for a solution and others. When someone would like to get an inspiration and advice, he may get some advice from tweets which are described suggestions for a solution which according to the wisdom of crowds [Surowiecki, 2005]. We conducted some experiments for tweet classification.

Thai language is a continuously written format. Therefore it is difficult to separate each word. Moreover, Twitter have limited the number of character a tweet and, free writing formatted, and users feel free to tweet, there are many ill-formed sentences. Therefore it is difficult to segment words.

As a result of the experiment, when we used Linear SVM, we obtained the best results. Moreover, we compared classification results due to the size of the corpus which was used for word embedding vector. At the incident: "Thai Cave Rescue" that happened in 2018 Jun-July, people from many countries tweeted and prayed for the safety of the boys trapped in the cave. We found that these tweets can be classified into the following four types.

- Suggesting solutions
- Emotional tweets
- News Reports
- Other tweets

The experimental results showed that we can obtain good results. And also we compared the results using tweets written in Thai language and the results using tweets written in Japanese.

4.3 Experimental Results

4.3.1 Experimental Setup

As the experimental data, we selected tweets about "Thai cave rescue (Jun-July 2018)". Table 4.1 illustrates the quantities of tweets written in Thai language and Japanese.

Table 4.1 Quantities of tweets about "Thai Cave Rescue"

Date	Thai Language	Japanese
July 2	2	3
July 3	10	21
July 4	20	20
July 5	24	17
July 6	38	6
July 7	69	5
July 8	8,512	1,065
July 9	7,820	865
July 10	9,823	915
July 11	5,706	909
July 12	1,311	416
Total	33,335	4,242

Although Twitter users in Thai are fewer than Japanese users (Thai: 12M users, Japanese: 40M users), the number of Thai tweets is larger than the number of Japanese tweets.

For tweet classification, we decide to classify the tweets into four classes. Table 4.2 illustrates four classes which are classified by our system.

Table 4.2 Four classes of tweets

Class	Description
Solution	tweets which include how to escape from the cave
Emotion	tweets which include user's emotion
Report	news report
Others	others

For classification experiments, we labeled 2,100 tweets written in Thai language. Table 4.3 shows the distribution of labeled tweets.

Table 4.3 Distribution of labeled Thai-tweets

Class	Number of tweets
Solution	158
Emotion	1,078
Report	560
Others	304
Total	2,100

Sometimes we want to add multiple labels to one tweet, but in this experiment, we decided to attach one label to each tweet instead of multi-label. For example, after quoting news a word a tweet that states his opinion is classified as Emotion instead of News Report. If it is like a news report but a concrete rescue method is described, Method is chosen as the label.

4.3.2 Our Classifier Framework

Our classifier utilize word embedding from word2vec as input data that concatenate a word vector of each word of a tweet to represent each tweet vector before feeds to the input of classifiers, that shows in Figure 4.2 below.

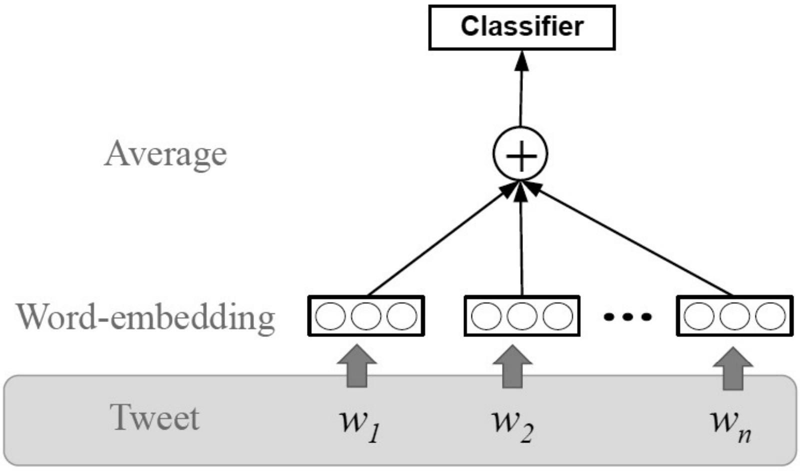


Fig. 4.2 Our Classifier Framework

4.3.3 Experimental Results

Each experiment was conducted with 10 fold cross-validation. For classification of tweet data, we employed five kinds of machine learning method: Linear SVM, two kinds of Naive Bayes (Gaussian and Bernoulli), Stochastic Gradient Descent and Passive Aggressive Classifier.

Table 4.4 illustrates the results of the experiment using the word embedding vector from the large corpus.

Table 4.4 classification result (f-score) corpus size: 696MB, vocab: 30,225

Method	Micro	Macro
SVM (Linear)	0.69	0.59
Naive Bayes (Gaussian)	0.56	0.49
Naive Bayes (Bernoulli)	0.62	0.55
Stochastic Gradient Descent	0.67	0.57
Passive Aggressive Classifier	0.63	0.55

Table 4.5 illustrates the results of the experiment using the word embedding vector from the small corpus.

Table 4.5 classification result (f-score) corpus size: 500MB, vocab: 60,002

Method	Micro	Macro
SVM (Linear)	0.71	0.61
Naive Bayes (Gaussian)	0.44	0.42
Naive Bayes (Bernoulli)	0.56	0.64
Stochastic Gradient Descent	0.57	0.41
Passive Aggressive Classifier	0.55	0.34

In both experiments, we obtained the best results when we used Linear SVM.

4.3.4 Discussion

Table 4.6, 4.7, 4.8, 4.9, 4.10 shows confusion matrix of the classification results by each classification method, respectively. In each Table, S, E, R, and O illustrates four classes: Solutions, Emotion, news Reports, and Others, respectively.

Table 4.6 confusion matrix of classification by Linear SVM (Thai-tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	7	3	4	0
by	E	3	86	10	8
hand	R	3	11	38	2
	O	1	13	4	11

Table 4.7 confusion matrix of classification by Naive Bayes(Gaussian) (Thai-tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	9	1	3	1
by	E	11	64	16	15
hand	R	12	6	31	5
	O	1	11	6	11

Table 4.8 confusion matrix of classification by Naive Bayes(Bernoulli) (Thai-tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	8	2	3	0
by	E	4	74	15	12
hand	R	9	11	31	3
	O	0	11	3	14

Table 4.9 confusion matrix of classification by Stochastic Gradient Descent (Thai-tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	6	3	4	0
by	E	1	87	8	10
hand	R	2	15	35	2
	O	1	15	2	10

Table 4.10 confusion matrix of classification by Passive Aggressive Classifier (Thai-tweets)

		Labeled by system			
		S	E	R	O
Labeled by hand	S	8	2	3	1
	E	3	74	10	18
	R	3	9	38	4
	O	1	13	4	10

We found that it is difficult to classify others. Because there are various types of tweets that are classified into ’Others’.

4.3.5 Comparison classification results between Thai and Japanese

We compared the classification results of Thai tweets and that of Japanese tweets.

Table 4.11 illustrates the average number of words per tweet written in Thai language and Japanese.

Table 4.11 the average number of words per a tweet written in Thai language and Japanese

Language	Thai	Japanese
# of tweets	2,100	2,021
# of vocabulary	5,071	4,902
# of word(token)	44,347	72,630
Average # of words(token) per tweet	21.12	35.94

The data show Japanese tweets are longer than Thai tweets.

For classification experiments, we labeled 2,021 tweets written in Japanese. Table 4.12 shows the distribution of labeled tweets.

Table 4.12 Distribution of labeled Japanese-tweets

Class	Number of tweets
Method	84
Emotion	490
Report	1,280
Others	167
Total	2,021

Table 4.14 illustrates the results against Japanese tweets by the same methods. We used two different word embedding vectors. The first model is from the site: "Pre-trained word vectors of 30+ languages" [Park, 2017]. The second model is made with Wikipedia data. These models are provided by word2vec [Mikolov et al., 2013a]. For morphological analysis of Japanese tweets, we used MeCab [Kudo et al., 2004]. The features of the models are as shown in Table 4.13.

Table 4.13 The features of the two Japanese word vector models

	Model	
	First model	Second model
Vector size	300	300
Corpus size	1GB	3.3GB
Vocabulary size	50,108	519,275

Table 4.14 illustrates the results of the experiment using the word embedding vector from the small corpus.

Table 4.14 classification result (f-score) corpus: 1GB, vocab: 50,108 (Japanese)

Method	Micro	Macro
SVM (Linear)	0.79	0.66
Naive Bayes (Gaussian)	0.66	0.57
Naive Bayes (Bernoulli)	0.69	0.57
Stochastic Gradient Descent	0.80	0.67
Passive Aggressive Classifier	0.80	0.66

Table 4.15 illustrates the results of the experiment using the word embedding vector from the large corpus.

Table 4.15 classification result (f-score) corpus: 3.3GB, vocab: 519,275 (Japanese)

Method	Micro	Macro
SVM (Linear)	0.82	0.68
Naive Bayes (Gaussian)	0.71	0.54
Naive Bayes (Bernoulli)	0.72	0.57
Stochastic Gradient Descent	0.81	0.67
Passive Aggressive Classifier	0.81	0.68

In both experiments, we obtained good results when we used Linear SVM, Stochastic Gradient Descent and Passive Aggressive Classifier. Moreover, we found the results of Japanese tweets are better than the results of Thai tweets. It is because in the experiments of Japanese tweets we used word embedding vectors from the big corpus.

Table 4.16, 4.17, 4.18, 4.19, 4.20 shows confusion matrix of the results by each classification method, respectively. In each Table, S, E, R, and O illustrates four classes: Solutions, Emotion, News Reports, and Others, respectively.

Table 4.16 confusion matrix of classification by Linear SVM (Japanese-Tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	4	1	1	0
by	E	0	34	12	1
hand	R	0	5	120	0
	O	0	6	4	5

Table 4.17 confusion matrix of classification by Naive Bayes(Gaussian) (Japanese-Tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	5	1	1	0
by	E	2	34	9	2
hand	R	6	18	100	2
	O	0	10	2	3

Table 4.18 confusion matrix of classification by Naive Bayes(Bernoulli) (Japanese-Tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	4	2	0	0
by	E	1	32	10	5
hand	R	4	16	103	3
	O	0	8	3	5

Table 4.19 confusion matrix of classification by Stochastic Gradient Descent (Japanese-Tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	4	1	1	0
by	E	0	34	11	2
hand	R	0	6	119	1
	O	0	5	4	6

Table 4.20 confusion matrix of classification by Passive Aggressive Classifier (Japanese-Tweets)

		Labeled by system			
		S	E	R	O
Labeled	S	5	1	1	0
by	E	1	31	13	2
hand	R	1	5	120	1
	O	0	4	3	7

We found that it is difficult to classify ’Emotion’ from tweet data. Some of ’Emotion’ tweets are classified as ’News Report’. It is because many Japanese users quoted news reports even if they would to like mention their emotion.

4.3.6 Conclusion

This paper describes how to classify Thai language tweets. In this study, we collected experimental data from Twitter data which have a specific hashtag. Even if tweet data which have the same hashtags, there are many kinds of tweets: suggesting solutions, emotional tweets, news report, and other tweets. When someone would like to get an inspiration and advice, he may get some advice from tweets which are described suggesting solutions. We conducted some experiments for tweet classification using five machine learning algorithm. The experimental results showed that we can obtain good results. Moreover, we compared the results using tweets written in Thai language and the results using tweets written in Japanese.

We conducted some classification experiments for Twitter data using five kinds of machine learning algorithm. When users post their feelings by Twitter, they often quote news articles before their emotional tweet. Therefore, it is difficult to improve classification accuracy with the method using only the frequency of words that was conventionally done.

In the experiments of Japanese tweet classification, We obtained better results than the results of the Thai language result. The possible reasons are as follows.

- Accuracy of morphological analysis
- Amount of corpus for word embedding vector
- number of words per tweet

Moreover, we were able to compare the contents of Thai and Japanese tweets against the same incident. We compared the results of Thai tweets and the results of Japanese tweets against the incident "Thai cave rescue" which occurred from June to July 2018. In the experiments we extracted many tweets which mentioned some ideas for rescuing children from the cave: e.g. drainage by air pumps, mini-submarine, air pumps & tubes and so on.

4.4 Related Works

There are many research papers for short text classification. For example, Alsmadi et al. proposed a term weighting scheme for short-text classification that called the supervised weight scheme [Alsmadi and Hoon, 2018] which obtained higher performance than traditional weighting i.e. term frequency(TF), binary-weight, term frequency-inverse document frequency(TF-IDF). In terms of the Thai language, Nomponkrang et al. presented a comparative study of classification algorithms for Thai-sentence into Four classes; interrogative, exclamatory, imperative and declarative by four classification algorithms these are Navie Bayes, k-NN, Decision Tree and Support Vector Machine (SVM) respectively. In pre-processing, for weighting input words in a sentence with term binary, TF and TF-IDF. [Nomponkrang and Sanrach, 2016]. And P. Sarakit et al. contributed the classifying emotion in Thai YouTube comments [Sarakit et al., 2015], shown the comparative result of SVM, Decision Tree and Multinomial Naive Bayes. And also, P. Vateekul and T. Koomsubha proposed the sentiment analysis of Thai Twitter data utilize deep learning techniques [Vateekul and Koomsubha, 2016], they are Long Short-Term Memory (LSTM) and Dynamic Convolutional Neural Network (DCNN), using word embedding from Word2vec as input data point for both of them. To due with unbalanced data classifying, W. Wunnasri et al. proposed the approach for solving unbalanced data for Thai tweets sentiment classification [Wunnasri et al., 2013].

Chapter 5

Conclusion and Future Works

5.1 Conclusion

Twitter data is a huge social media data source. It can reflect the opinions and information about the events or mention to new technologies from all over the world, almost like real-time data. Many researchers have paid attention to the Twitter data for their study and research.

On the other hand, the disadvantage of the tweet is that short messages are limited to a maximum of 140 characters. Currently, there are expanded to 280 characters for some languages. Additional, Additional, Twitter users can free written style (informally written format), which effect to tweets have slang, emoji, and unknown words are included a lot. As a result, the extraction of information from Twitter data is difficult and challenging.

The first challenge, we studied for solving a problem of ambiguous words in the tweets, especially related to the technology domain. Human beings solve the problem of the ambiguous word by using its context to choose the right meaning for understanding the meaning of the texts correctly. Similarly, computer algorithms also rely on contextual information. The main issue of Twitter data is tweets are a short text, so the ambiguous words in tweets do not have enough context information to decide the right meaning. Therefore, we introduced a way to extend the context of the ambiguous word in Twitter data by taking advantage of the word embedding (word vector) which was created by word2vec program, the word embedding of each word, based on its context data encoding. Experimental results showed that the extended context approach is the better way to address the ambiguity in tweets. In the future, we want to exceed the performance of our proposed method.

The second challenge is unknown words in Twitter data that may be a new word which was created to describe the emerging technologies. We presented a method for detecting new-words in Twitter data which do not appear in the WordNet resource files and defining their sense utilized abstract information from Wikipedia. To definition new word sense, our proposed method utilize word embedding from word2vec to select the most meaningful sentence from Wikipedia abstract information page of an each new word. Experimental results were satisfactory, however we have to further study for improving our proposed method performance.

In the last section, we have studied to extracting semantic information from the specific incident Twitter data which the same hash-tag. We using Twitter data of the Thai cave rescue incident during July 2 – July 10, 2018. These tweets can be classified into four classes consist of news report tweet, proposed solution tweet, emotions tweet, and others tweet. In our experiments, to classify Thai tweets using existing algorithms (machine learning algorithm) and feed the word embedding as input of classifier. We also compare with the Japanese tweets classification. The results have shown that the SVM classifier provides the best results. The obtained classified information can be used to create a timeline of events and can be summarized key proposed solution ideas of the incident.

5.2 Future Works

Due to the continuous development of technology, which effects on human society and effects on the language used in communication. In order to describe those new technologies, the new words have been composed constantly.

In natural language processing domain, there are many tasks require the knowledge-based, such as WordNet. That cannot be added new-words up to date frequently because of relatively expensive. In future work, we intend to use our system to collects new words and meanings them and after that create an append file for use with WordNet database files that can be used for further natural language processing.

Twitter data is a challenging new source for semantic knowledge mining. In our work involved the extracting semantic information of incident from Twitter data. In future work, we expect to create each incident ontology (semantic graph). That can be used for comparative studies for future events such as pestilences, disasters, North Korea missiles, etc.

References

E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of NAACL-HLT*, 2009.

I. Alsmadi and G. K. Hoon. Term weighting scheme for short-text classification: Twitter corpuses. *Neural Computing and Applications*, page 1–13, 2018.

T. Aoki, K. Yoshikawa, T. Nasukawa, H. Takamura, and M. Okumura. Detecting earthquake survivors with serious mental affliction. In *Proceedings of 2017 Pacific Association for Computational Linguistics*, 2017.

S. Banerjee and T. Extended gloss overlaps as a measure of semantic relatedness. In *Proceeding IJCAI'03 Proceedings of the 18th international joint conference on Artificial intelligence*, pages 805–810, Acapulco, Mexico, 2003.

R. Barzilay and K.R. McKeown. Extracting paraphrases from a parallel corpus. In *In Proc. of 39th Annual Meeting of the Association for Computational Linguistics*, pages 50–57, 2001.

J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

L. Chin-Yew. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop, vol. 8.*, 2004.

Google code. Word2vec code, 2013. URL https://code.google.com/p/word2vec/.

W. Dingding, S. Sahar, and L. Tao. Update summarization using semi-supervised learning based on hellinger distance. In *In CIKM'15 Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1907–1910, 2015.

C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, 2002.

D. Hindle. Noun classification from predicate argument structures. In *Proceedings of 28th Annual Meeting of the Association for Computational Linguistics*, pages 268–275, 1990.

M. Huang, B. Ye, Y. Wang, H. Chen, J. Cheng, and X. Zhu. New word detection for sentiment analysis. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 531–541, 2014.

J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*, Taiwan, 1997.

N. Kanungsukkasem and T. Leelanupab. Correlations between twitter data and financial instruments. *KMITL Journal of Information Technology*, 3(2), 2014.

T. Kudo, K. Yamamoto, and Y. Matsumoto. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, page 230–237, Barcelona, Spain, 2004.

M. J. Kusner, Y. Sun, N. I. Kolkin, and K. Q. Weinberger. From word embeddings to document distances. In *Proc. of the 32nd International Conference on Machine Learning*, pages 957–966, 2015.

C. Leacock and M. Chodorow. Combining local context andwordnet similarity for word sense identification. *In C. Fellbaum, editor, WordNet: An electronic lexical database*, page 265–283, 1998.

D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 768–773, 1998a.

D. Lin. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning*, Madison, Wisconsin, 1998b.

D. Lin and P. Pantel. Discovery of inference rules for question answering. *Natural Language Engineering*, 7, No.4:343–360, 2001.

A. Mehwish and R. Muhanmmad. Sentence based semantic similarity measure for blog-posts. *6th International Conference on Digital Content, Multimedia Technology and its Applications*, pages 69–74, 2010.

L. Meng, R. Huang, and J. Gu. A review of semantic similarity measures in wordnet. *International Journal of Hybrid Information Technology*, volume 6, 2013.

D. Metzler, E. Hovy, and C. Zhang. An empirical evaluation of data-driven paraphrase generation techniques. In *In Proc. of 49th Annual Meeting of the Association for Computational Linguistics*, pages 50–57, 2011.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*, 2013a.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26, pages 3111–3119, 2013b.

G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

T. Nomponkrang and C. Sanrach. The comparison of algorithms for thai-sentence classification. *International Journal of Information and Education Technology*, Vol. 6:801–808, 2016.

K. Park. Pre-trained word vectors of 30+ languages, 2017. URL https://github.com/Kyubyong/wordvector.

S. Patwardhan and T. Pedersen. Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In *Proc. of 11th conference of the European Chapter of the Association for Computational Linguistics(EACL-2006)*, Trento, Italy, 2006.

T. Pedersen, S. Patwardhan, and J. Michelizzi. Wordnet::similarity-measuring the relatedness of concepts. *Association for computational Linguistics–HLT-NAACL*, pages 38–41, 2004.

C. Quirk, C. Brockett, and W. Dolan. Monolingual machine translation for paraphrase generation. In *In Proc. of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 142–149, 2004.

R. Rada, H. Mili, E. Bicknell, and M. Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30, 1989.

P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Montreal, 1995.

M. A. Rodriguez and M. J. Egenhofer. Determining semantic similarity among entity classes from different ontologies. *IEEE Transactions on Knowledge and Data Engineering*, vol. 15 No.2, 2003.

G. Salton and C. Buckley. The smart retrieval system experiments in automatic text retrieval. *Information processing & management*, vol. 24:513–523, 1988.

P. Sarakit, T. Theeramunkong, C. Haruechaiyasak, and M. Okumura. Classifying emotion in thai youtube comments. In *Proceedings of the 6th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*, 2015.

H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, 1994.

scikit learn. Naive bayes[online available], 2018a. URL https://scikit-learn.org/stable/modules/naive_bayes.html.

scikit learn. Passive aggressive classifier[online available], 2018b. URL https://scikit-learn.org/stable/modules/linear_model.html.

scikit learn. Stochastic gradient descent[online available], 2018c. URL https://scikit-learn.org/stable/modules/sgd.html.

James Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.

S. Takaleand and S. Nandgaonkar. Measuring semantic similarity between words using web documents. *International Journal of Advanced Computer Science and Applications*, 1(4): 78–85, 2010.

R. Thomson, N. Ito, H. Suda, F. Lin, Y. Liu, R. Hayasaka, R. Isochi, and Z. Wang. Trusting tweets: The fukushima disaster and information source credibility on twitter. In *Proceeding of the 9th International ISCRAM Conference – Vancouver, Canada*, 2012.

P. Vateekul and T. Koomsubha. A study of sentiment analysis using deep learning techniques on thai twitter data. In *Proceedings of the 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2016.

Wikipedia. Technology [online available], 2018. URL https://en.wikipedia.org/wiki/Technology.

H. Wilensky. Twitter as a navigator for stranded commuters during the great east japan earthquake. In *Proceedings of the 11th International ISCRAM Conference – University Park, Pennsylvania, USA*, 2014.

Z. Wu and M. Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd Meeting of Association of Computational Linguistics*, page 33–138, 1994.

W. Wunnasri, T. Theeramunkong, and C. Haruechaiyasak. Solving unbalanced data for thai sentiment. In *Proceedings of the 10th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 200–205, 2013.

Appendix A

DATA RELATED TO RESEARCH

30 ambiguous words used for evaluation

In our study, we selected 30 ambiguous words related to technology domain for evaluating the word sense identification experiments with Twitter data that shown as the table below.

Table A.1 The ambiguous words and their number of sense in WordNet3.1

Target word	# senses	Target word	# senses	Target word	# senses
ai	4	memory	5	protocol	3
application	7	method	2	router	3
cloud	14	mouse	6	screen	15
cookie	3	notebook	2	signal	6
desktop	2	ontology	2	spam	3
device	5	os	5	speaker	3
display	8	phone	4	vehicle	4
drone	7	platform	5	web	8
energy	7	processor	3	window	8
hacker	3	program	10	worm	5

The technology keywords used for getting Twitter data

we collected Twitter data via Twitter search API with specific keywords for achieving tweets in a specific domain which related to technology.

Table A.2 List of keywords which related to technology domain

Keywords used for Twitter search API		
4G	AI	android
Apple	apps	artificial intelligence
automobile	autonomous car	autonomous transportation
Bluetooth	bots	camera
communication	computer	computing
CPU	GIS	cyber security
cyber	digital	display
driverless	drone	earphone
electric vehicle	energy	engineering
engine	evolution	edge
flying robot	geo	hacker
handset	Hi-Fi	high-tech
hitech	hi-tech	hologram
human flying	humanoid robot	Information
innovation	Intel	intelligent vehicle
internet of things	logistics	machine
Microsoft	mobile phone	mobile
mobile robot	nanotechnology	NLP
node	OLED	ontology
periscope	research	robot
robotic	samsung	science
scientific	self driving	simulation
smart phone	smart	software
solar	solution	system
Tablet computer	technology	telepresence
smartwatch	touchpad	vehicle
visualization	WiFi	GPS

Stop words

There are many lists of stop-words available. We collected several stop-words lists then we put it together and sorts it by its frequency in WordNet glossary, in our experiment, we selected top frequency 60 words that show in a table as below.

Table A.3 List of stop words

the	who	especially	other
a	having	relating	their
of	used	something	all
or	he	usually	this
in	was	she	more
and	his	flowers	were
to	at	her	after
an	not	its	had
that	be	some	many
with	are	made	been
by	any	has	only
for	which	between	such
is	into	they	most
as	it	when	these
on	united	but	also

RG65 dataset

The 65 noun pairs were annotated by 51 human subjects, collected by Rubenstein and Goodenough (1965). All of the noun pairs are non-technical words and they are scored using a scale from 0 (not related) to 4 (perfect synonymy).

Table A.4 65 English noun word-pairs with human-assigned similarity judgments score

word-pair	rating score	word-pair	rating score
gem jewel	3.94	midday noon	3.94
automobile car	3.92	cemetery graveyard	3.88
cushion pillow	3.84	boy lad	3.82
cock rooster	3.68	implement tool	3.66
forest woodland	3.65	coast shore	3.60
autograph signature	3.59	journey voyage	3.58
serf slave	3.46	grin smile	3.46
glass tumbler	3.45	cord string	3.41
hill mound	3.29	magician wizard	3.21
furnace stove	3.11	asylum madhouse	3.04
brother monk	2.74	food fruit	2.69
bird cock	2.63	bird crane	2.63
oracle sage	2.61	sage wizard	2.46
brother lad	2.41	crane implement	2.37
magician oracle	1.82	glass jewel	1.78
cemetery mound	1.69	**car journey**	**1.55**
hill woodland	1.48	crane rooster	1.41
furnace implement	1.37	coast hill	1.26
bird woodland	1.24	shore voyage	1.22
cemetery woodland	1.18	food rooster	1.09
forest graveyard	1.00	lad wizard	0.99
mound shore	0.97	automobile cushion	0.97
boy sage	0.96	monk oracle	0.91
shore woodland	0.90	grin lad	0.88
coast forest	0.85	asylum cemetery	0.79
monk slave	0.57	cushion jewel	0.45
boy rooster	0.44	glass magician	0.44
graveyard madhouse	0.44	asylum monk	0.39
asylum fruit	0.19	grin implement	0.18
mound stove	0.14	automobile wizard	0.11
autograph shore	0.06	fruit furnace	0.05
noon string	0.04	rooster voyage	0.04
cord smile	0.02		

MC30 dataset

Miller and Charles repeated the experiments using a subset of RG65 dataset consists of 30 word-pairs. The relatedness of each word pair was rated by 38 human subjects rating scale from 0 to 4. [Miller and Charles, 1991]

Table A.5 List of MC30 word-pairs similar

word-pair	rating score	word-pair	rating score
car automobile	3.92	gem jewel	3.84
journey voyage	3.84	boy lad	3.76
coast shore	3.70	asylum madhouse	3.61
magician wizard	3.50	midday noon	3.42
furnace stove	3.11	food fruit	3.08
bird cock	3.05	bird crane	2.97
tool implement	2.95	brother monk	2.82
lad brother	1.66	crane implement	1.68
journey car	**1.16**	monk oracle	1.10
cemetery woodland	0.95	food rooster	0.89
coast hill	0.87	forest graveyard	0.84
shore woodland	0.63	monk slave	0.55
coast forest	0.42	lad wizard	0.42
chord smile	0.13	glass magician	0.11
rooster voyage	0.08	noon string	0.008

EN-WS-353 dataset

The WordSimilarity-353 Test Collection is English word pairs along with human-assigned similarity judgements [Finkelstein et al., 2002]. All the subjects (29 subjects) possessed near-native command of English. Their instructions were to estimate the relatedness of the words in pairs on a scale from 0 (totally unrelated words) to 10 (very closely related).

Agirre et al. divided WS-353 into two subsets, one for evaluating similarity (EN-WS-353-SIM), and the other for evaluating relatedness (EN-WS-353-REL) [Agirre et al., 2009].

EN-WS-353-SIM

Table A.6 Subset of EN-WS-353: similarity (1)

word-pair	rating score	word-pair	rating score
tiger cat	7.35	tiger tiger	10.00
plane car	5.77	train car	6.31
television radio	6.77	media radio	7.42
bread butter	6.19	cucumber potato	5.92
doctor nurse	7.00	professor doctor	6.62
student professor	6.81	smart stupid	5.81
wood forest	7.73	money cash	9.15
king queen	8.58	king rook	5.92
bishop rabbi	6.69	fuck sex	9.44
football soccer	9.03	football basketball	6.81
football tennis	6.63	Arafat Jackson	2.50
physics chemistry	7.35	vodka gin	8.46
vodka brandy	8.13	drink eat	6.87
car automobile	8.94	gem jewel	8.96
journey voyage	9.29	boy lad	8.83
coast shore	9.10	asylum madhouse	8.87
magician wizard	9.02	midday noon	9.29
furnace stove	8.79	food fruit	7.52
bird cock	7.10	bird crane	7.38
food rooster	4.42	money dollar	8.42
money currency	9.04	tiger jaguar	8.00
tiger feline	8.00	tiger carnivore	7.08
tiger mammal	6.85	tiger animal	7.00
tiger organism	4.77	tiger fauna	5.62
psychology psychiatry	8.08	psychology science	6.71
psychology discipline	5.58	planet star	8.45
planet moon	8.08	planet sun	8.02
precedent example	5.85	precedent antecedent	6.04
cup tableware	6.85	cup artifact	2.92
cup object	3.69	cup entity	2.15
jaguar cat	7.42	**jaguar car**	**7.27**
mile kilometer	8.66	skin eye	6.22
Japanese American	6.50	century year	7.59
announcement news	7.56	doctor personnel	5.00

Table A.7 Subset of EN-WS-353: similarity (2)

word-pair	rating score	word-pair	rating score
Harvard Yale	8.13	hospital infrastructure	4.63
life death	7.88	travel activity	5.00
type kind	8.97	street place	6.44
street avenue	8.88	street block	6.88
cell phone	7.81	dividend payment	7.63
calculation computation	8.44	profit loss	7.63
dollar yen	7.78	dollar buck	9.22
phone equipment	7.13	liquid water	7.89
marathon sprint	7.47	seafood food	8.34
seafood lobster	8.70	lobster food	7.81
lobster wine	5.70	championship tournament	8.36
man woman	8.30	man governor	5.25
murder manslaughter	8.53	opera performance	6.88
Mexico Brazil	7.44	glass metal	5.56
aluminum metal	7.83	rock jazz	7.59
museum theater	7.19	shower thunderstorm	6.31
monk oracle	5.00	cup food	5.00
journal association	4.97	street children	4.94
car flight	4.94	space chemistry	4.88
situation conclusion	4.81	word similarity	4.75
peace plan	4.75	consumer energy 4.75	
ministry culture	4.69	smart student	4.62
investigation effort	4.59	image surface	4.56
life term	4.50	start match	4.47
computer news	4.47	board recommendation	4.47
lad brother	4.46	observation architecture	4.38
coast hill	4.38	deployment departure	4.25
benchmark index	4.25	attempt peace	4.25
consumer confidence	4.13	start year	4.06
focus life	4.06	development issue	3.97
theater history	3.91	situation isolation	3.88
profit warning	3.88	media trading	3.88
chance credibility	3.88	precedent information	3.85
architecture century	3.78	population development	3.75
stock live	3.73	peace atmosphere	3.69
morality marriage	3.69	minority peace	3.69

Table A.8 Subset of EN-WS-353: similarity (3)

word-pair	rating score	word-pair	rating score
atmosphere landscape	3.69	report gain	3.63
music project	3.63	seven series	3.56
experience music	3.47	school center	3.44
five month	3.38	announcement production	3.38
morality importance	3.31	money operation	3.31
delay news	3.31	governor interview	3.25
practice institution	3.19	century nation	3.16
coast forest	3.15	shore woodland	3.08
drink car	3.04	president medal	3.00
prejudice recognition	3.00	viewer serial	2.97
peace insurance	2.94	Mars water	2.94
media gain	2.88	precedent cognition	2.81
announcement effort	2.75	line insurance	2.69
crane implement	2.69	drink mother	2.65
opera industry	2.63	volunteer motto	2.56
listing proximity	2.56	precedent collection	2.50
cup article	2.40	sign recess	2.38
problem airport	2.38	reason hypertension	2.31
direction combination	2.25	Wednesday news	2.22
glass magician	2.08	cemetery woodland	2.08
possibility girl	1.94	cup substance	1.92
forest graveyard	1.85	stock egg	1.81
month hotel	1.81	energy secretary	1.81
precedent group	1.77	production hike	1.75
stock phone	1.62	holy sex	1.62
stock CD	1.31	drink ear	1.31
delay racism	1.19	stock life	0.92
stock jaguar	0.92	monk slave	0.92
lad wizard	0.92	sugar approach	0.88
rooster voyage	0.62	noon string	0.54
chord smile	0.54	professor cucumber	0.31
king cabbage	0.23		

EN-WS-353-REL

Table A.9 Subset of EN-WS-353: relatedness (1)

word-pair	rating score	word-pair	rating score
computer keyboard	7.62	Jerusalem Israel	8.46
planet galaxy	8.11	canyon landscape	7.53
OPEC country	5.63	day summer	3.94
day dawn	7.53	country citizen	7.31
planet people	5.75	environment ecology	8.81
Maradona football	8.62	OPEC oil	8.59
money bank	8.50	computer software	8.50
law lawyer	8.38	weather forecast	8.34
network hardware	8.31	nature environment	8.31
FBI investigation	8.31	money wealth	8.27
psychology Freud	8.21	news report	8.16
war troops	8.13	physics proton	8.12
bank money	8.12	stock market	8.08
planet constellation	8.06	credit card	8.06
hotel reservation	8.03	closet clothes	8.00
soap opera	7.94	planet astronomer	7.94
planet space	7.92	movie theater	7.92
treatment recovery	7.91	baby mother	7.85
money deposit	7.73	television film	7.72
psychology mind	7.69	game team	7.69
admission ticket	7.69	Jerusalem Palestinian	7.65
Arafat terror	7.65	boxing round	7.61
computer internet	7.58	money property	7.57
tennis racket	7.56	telephone communication	7.50
currency market	7.50	psychology cognition	7.48
seafood sea	7.47	book paper	7.46
book library	7.46	psychology depression	7.42
fighting defeating	7.41	movie star	7.38
hundred percent	7.38	dollar profit	7.38
money possession	7.29	cup drink	7.25
psychology health	7.23	summer drought	7.16
investor earning	7.13	company stock	7.08
stroke hospital	7.03	liability insurance	7.03
game victory	7.03	psychology anxiety	7.00

Table A.10 Subset of EN-WS-353: relatedness (2)

word-pair	rating score	word-pair	rating score
game defeat	6.97	FBI fingerprint	6.94
money withdrawal	6.88	psychology fear	6.85
drug abuse	6.85	concert virtuoso	6.81
computer laboratory	6.78	love sex	6.77
problem challenge	6.75	movie critic	6.73
Arafat peace	6.73	bed closet	6.72
lawyer evidence	6.69	fertility egg	6.69
precedent law	6.65	minister party	6.63
psychology clinic	6.58	cup coffee	6.58
water seepage	6.56	government crisis	6.56
space world	6.53	dividend calculation	6.48
victim emergency	6.47	luxury car	6.47
tool implement	6.46	competition price	6.44
psychology doctor	6.42	gender equality	6.41
listing category	6.38	video archive	6.34
oil stock	6.34	governor office	6.34
discovery space	6.34	record number	6.31
brother monk	6.27	production crew	6.25
nature man	6.25	family planning	6.25
disaster area	6.25	food preparation	6.22
preservation world	6.19	movie popcorn	6.19
lover quarrel	6.19	game series	6.19
dollar loss	6.09	weapon secret	6.06
shower flood	6.03	registration arrangement	6.00
arrival hotel	6.00	announcement warning	6.00
game round	5.97	baseball season	5.97
drink mouth	5.96	life lesson	5.94
grocery money	5.94	energy crisis	5.94
reason criterion	5.91	equipment maker	5.91
cup liquid	5.90	deployment withdrawal	5.88
tiger zoo	5.87	**journey car**	**5.85**
money laundering	5.65	summer nature	5.63
decoration valor	5.63	Mars scientist	5.63
alcohol chemistry	5.54	disability death	5.47
change attitude	5.44	arrangement accommodation	5.41
territory surface	5.34	size prominence	5.31

Table A.11 Subset of EN-WS-353: relatedness (3)

word-pair	rating score	word-pair	rating score
exhibit memorabilia	5.31	credit information	5.31
territory kilometer	5.28	death row	5.25
doctor liability	5.19	impartiality interest	5.16
energy laboratory	5.09	secretary senate	5.06
death inmate	5.03	monk oracle	5.00
cup food	5.00	journal association	4.97
street children	4.94	car flight	4.94
space chemistry	4.88	situation conclusion	4.81
word similarity	4.75	peace plan	4.75
consumer energy	4.75	ministry culture	4.69
smart student	4.62	investigation effort	4.59
image surface	4.56	life term	4.50
start match	4.47	computer news	4.47
board recommendation	4.47	lad brother	4.46
observation architecture	4.38	coast hill	4.38
deployment departure	4.25	benchmark index	4.25
attempt peace	4.25	consumer confidence	4.13
start year	4.06	focus life	4.06
development issue	3.97	theater history	3.91
situation isolation	3.88	profit warning	3.88
media trading	3.88	chance credibility	3.88
precedent information	3.85	architecture century	3.78
population development	3.75	stock live	3.73
peace atmosphere	3.69	morality marriage	3.69
minority peace	3.69	atmosphere landscape	3.69
report gain	3.63	music project	3.63
seven series	3.56	experience music	3.47
school center	3.44	five month	3.38
announcement production	3.38	morality importance	3.31
money operation	3.31	delay news	3.31
governor interview	3.25	practice institution	3.19
century nation	3.16	coast forest	3.15
shore woodland	3.08	drink car	3.04
president medal	3.00	prejudice recognition	3.00
viewer serial	2.97	peace insurance	2.94
Mars water	2.94	media gain	2.88

Table A.12 Subset of EN-WS-353: relatedness (4)

word-pair	rating score	word-pair	rating score
precedent cognition	2.81	announcement effort	2.75
line insurance	2.69	crane implement	2.69
drink mother	2.65	opera industry	2.63
volunteer motto	2.56	listing proximity	2.56
precedent collection	2.50	cup article	2.40
sign recess	2.38	problem airport	2.38
reason hypertension	2.31	direction combination	2.25
Wednesday news	2.22	glass magician	2.08
cemetery woodland	2.08	possibility girl	1.94
cup substance	1.92	forest graveyard	1.85
stock egg	1.81	month hotel	1.81
energy secretary	1.81	precedent group	1.77
production hike	1.75	stock phone	1.62
holy sex	1.62	stock CD	1.31
drink ear	1.31	delay racism	1.19
stock life	0.92	stock jaguar	0.92
monk slave	0.92	lad wizard	0.92
sugar approach	0.88	rooster voyage	0.62
noon string	0.54	chord smile	0.54
professor cucumber	0.31	king cabbage	0.23

Appendix B

PUBLISHED PAPER

(1) Apichai Chan-Udom, Chan Karman and Yoshimi Suzuki, *Identification of Word Sense in Twitter Data Based on WordNet Glosses*, In Proc. of 15th International conference of the Pacific Association for Computational Linguistics, August 2017, Yangon, Myanmar.

(2) Apichai Chan-Udom, Chan Karman and Yoshimi Suzuki, *Detection of new words and their senses in Twitter data using Wikipedia*, In Proc. of 8th Language and Technology Conference (LTC'17), November 2017, Poznań, Poland.

(3) Apichai Chan-Udom, Chan Karman and Yoshimi Suzuki, *Classification of Thai Tweets: Mining Treasures from Tweet Heap*, In Proc. of 5th International Conference on Systems and Informatics (ICSAI 2018), November 2018, Nanjing, China.

(4) Apichai Chan-Udom, Chan Karman and Yoshimi Suzuki, *Classifying Short Text in Social Media for Extracting Valuable Ideas*, In Proc. of 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2019), April 2019, La Rochelle, France.