TOWSON UNIVERSITY

OFFICE OF GRADUATE STUDIES

SEARCH KEYWORD SELECTION FOR CRAWLING TWEETS BASED ON THE KEYWORD EXTRACTION

by

Jeongwoo Kim

A Thesis

Presented to the faculty of

Towson University

in partial fulfillment

of the requirements for the degree

Master of Science in Information Technology

Department of Computer and Information Sciences

Towson University

Towson, Maryland 21252

December 2015

TOWSON UNIVERSITY
OFFICE OF GRADUATE STUDIES

THESIS APPROVAL PAGE

This is to certify that the thesis prepared by Jeongwoo Kim

entitled

Search Keyword Selection for Crawling Tweet using The Keyword Extraction

has been approved by the thesis committee as satisfactorily completing the thesis requirements for the degree MS in CS

Chairperson, Thesis Committee Signature	YANGGON KIM	11/23/2015
Committee Member Signature	Michael McGuire	11/23/2015
Committee Member Signature	Nam Nguyen	11/23/2015
Committee Member Signature	Type Name	Date
Committee Member Signature	Type Name	Date
Dean of Graduate Studies	Janet V. DeLany	12-9-15

Type Name / Date

Acknowledgement

I would like to express the deepest appreciation to my committee chairperson, Dr. Yanggon Kim, who has given me great advice on this research. Also, I would like to thank my committee members, Dr. Nam Nguyen and Dr. Michael McGuire for their valuable discussions.

I also would like to thank colleagues, Youngsub Han, Beomseok Hong, and Jinhyuck Choi. Their support on this research has been great encouragement to me.

ABSTRACT

SEARCH KEYWORD SELECTION FOR CRAWLING TWEETS BASED ON THE KEYWORD EXTRACTION

Jeongwoo Kim

As use of Social Network Services (SNSs) has been increased over and over, demands to derive meaningful information from them are continuing. In order to extract meaningful information from SNSs, to collect data from them should come up on a first step. In the data collection, keyword-based search is widely used to collect data from SNSs using Application Programming Interface (API).

However, in this data collection, a lot of extraneous data can be collected according to a selected topic. For example, if using the topic term such as “Coach” (Fashion company) as a search keyword, extraneous data unrelated to the topic are collected as well because term “Coach” is homonym. This problem makes the data analysis more difficult and causes a waste of data storage space. Additionally, it causes a waste of limited resources to collect data such as search queries.

For the topics in which the topic term is homonym, more terms for search keywords must be needed in order to collect data more accurately. Also, the terms should be extracted based on the real data. In this thesis, we propose a method to extract search keywords to be effective for collecting data related with a topic using tweets.

Table of Contents

List of Tables

List of Figures

1. Introduction

Continuous developments of mobile devices and the network technology over the last decade have made people use the internet anywhere and in anytime. Social media typically used on the internet by a lot of people has diversified and increased in number. Especially, Social Networking Services (SNSs) such as Twitter, Facebook, and Instagram have been in the spotlight. On various SNSs, many people create their accounts, communicate with other people and share their daily life, interests, opinions and news. Figure 1 shows an increase in the number of SNS users worldwide from 2010 to 2014 with predictions from 2015 to 2018 [1].

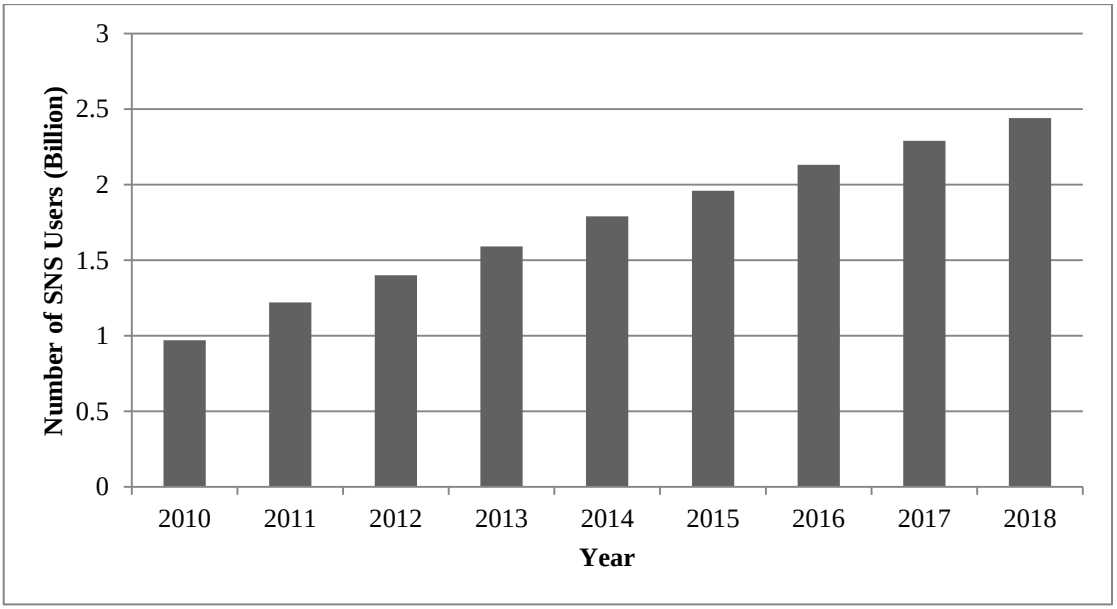

Figure 1. Increase in Number of User of SNS worldwide (In billion)

Also, for popularity of SNSs, A lot of companies consider SNSs as their good marketing tools. They easily can advertise their new products and events in widely connected social network using SNSs.

In this trend, a volume of data generated on SNSs is becoming larger day after day, and consequently demands to derive meaningful information from them are continuing. Meaningful information extracted from data generated on SNSs is widely

used not only for event detection, prediction of an election, influence of a user on social network, online reputation of people, products, or companies but also enterprise decision-making [2][3][4].

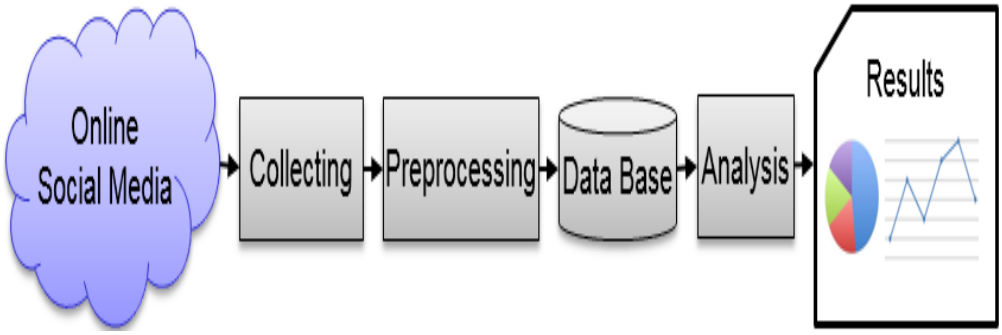

Figure 2. General Process to Extract Meaningful Information from SNSs

Generally, a process to extract meaningful information from data generated on SNSs can be summarized as shown in Figure 2 [5]. For a preprocessing step in which raw data collected from SNSs are transformed into a format for effective analysis of the data and analysis step in which various data mining techniques are applied to the transformed data to discover meaningful information, the data collection step is needed first.

Some SNSs allow developers to use their data by providing Application programming interfaces (APIs). Using the API, various ways to collect data can be used according to SNS.

The data collection using keyword-based search is a widely used way among the various ways to collect data from SNSs. In the data collection using keyword-based search, to determine search keywords is a very important. Generally, topic terms that represent topics such as companies and products are specified as search keywords. For example, if selecting topics such as Google and Verizon, “Google” and “Verizon” are given as search keywords to collect data. However, sometimes, this simple task is not easy because an unexpected collection of extraneous data can be caused according to

a topic. For example, if specifying “Target” and “Sonata” as search keywords after choosing topics such as Target (a retailing company) and Sonata (a car made by Hyundai), a lot of extraneous data are collected together because the terms, “Target” and “Sonata”, are homonyms sharing the same spelling and pronunciation with different meanings. This problem provokes a waste of data storage space and makes analysis of data more difficult. When handling data consisting of more than terabytes or petabytes, it will be critical in a system. Also, it escalates in that resources to collect data are limited in APIs. As mentioned, SNSs allow using their data by providing APIs, but it has some limitations of various types. For example, using keyword-based search with Twitter API, in order to collect tweets, messages written by users with 140 characters or less in Twitter, we can use 180 search queries per 15 minutes. Also, tweets searched by the given search queries are limited to 100 per a search query. In this condition, the problem that a lot of extraneous data can be collected according to a topic causes inefficient use of the limited resources.

Thus, efforts to collect data more accurately in keyword-based search for topics the topics in which the topic term is homonym are needed. If using more terms relevant to a particular topic with the topic term such as “Target” and “Sonata” as multiple search keywords, we can present the topic to collect more obviously. Therefore, we can expect to collect data relevant with the topic more accurately.

In this thesis, we propose a method to extract search keywords to be effective for crawling tweets relevant to a particular topic using the keyword extraction methods. As tweets are used for various analyses in a lot of researches, we chose to collect tweets.

This thesis is organized as follows. Section 2 contains a description of related works. In section 3, we explain a method to extract search keywords for crawling

tweets in detail. The experiment result is described in Section 4, And finally, we concludes with summarization.

2. Related Work

In this section, we explain some techniques for the keyword extraction. In the first section, we present basic concepts and applications of the keyword extraction, and then we explain Term Frequency-Inverse Document Frequency (TF-IDF) weighing scheme and Pointwise Mutual Information (PMI) in detail. In the second section, we describe Natural Language Processing (NLP).

2.1. Keyword Extraction

Keyword extraction is an important and essential technique for text mining such as information retrieval, text categorization, summarization and topic detection. A set of keywords extracted from a large-scale electronic document data are used for significant features to improve the performance of text mining. Approaches for keyword extraction can be divided into the four categories, simple statistics, linguistics, machine learning and other approaches [6].

Statistics approaches consisting of simple methods do not require the training data unlike machine learning approaches. These approaches are generally based on statistical information derived from the corpus and independent of languages and domains. These approaches include term frequency, TF-IDF, co-occurrences, and so on.

Linguistics approaches use the linguistics features and knowledges of the terms. Thus, these approaches need knowledge of domain and language. Lexical, syntactic, semantic and discourse analysis are common.

Machine learning approaches employ supervised or unsupervised learning. These approaches are usually based on supervised learning. In these approaches, the

keywords extracted from the training data are used to learn a model, and then the learned model is applied for the keyword extraction for new data. These approaches include Naïve Bayes, Support Vector Machine (SVM), Decision tree, Bagging, etc.

Other Approaches for the keyword extraction mainly combine the methods mentioned above or use some heuristic knowledge in the task of keyword extraction, such as the position, length, layout feature of the words, html tags around of the words, etc.

2.1.1. Term Frequency-Inverse Document Frequency

The TF-IDF (Term Frequency-Inverse Document Frequency) is a widely used weighing scheme in information retrieval and text mining. The weighing scheme weights a given term to determine how well the term describes an individual document within a corpus [7]. In other words, the idea is to find terms frequently occurring in a particular document while not occurring in other documents in a corpus.

The TF (Term Frequency) of a term means the number of times the term appears in a document. The simple scheme to determine TF value is raw frequency of a term in a document. There are some variations for determining TF value [8][9].

The IDF (Inverse Document Frequency) is a measure of the general importance of a term [9]. It is based on the DF (Document Frequency). The DF of a term means the number of documents in which the term appears in a corpus. Also, there are variations for determining IDF value [9].

The conventional TF-IDF weight scheme is defined as [8]:

$$tfidf_{i,j} = tf_{i,j} \text{ x } idf_i$$

TF value is raw frequency normalized by the number of times all terms appears in a document to prevent a bias towards longer documents. TF value is calculated as:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Where $n_{i,j}$ is the number of times term i in document d_j and $\sum_k n_{k,j}$ is the number of time all terms appears in document d_j. Also, IDF of a term can be defined as:

$$idf_i = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Where $|\{d \in D : t \in d\}|$ is the number of documents in which term *t* appears and N is total number of all documents in a corpus.

2.1.2. Pointwise Mutual Information

The Pointwise Mutual Information (PMI) introduced by Church and Hanks (1990) [10] is a measure of association between particular events x and y. PMI score of two events x and y is defined as:

$$pmi(x,y) = \log \frac{p(x,\ y)}{p(x)p(y)}$$

Where $p(x)$ is the probability that the event x occurs, $p(y)$ is the probability that the event y occurs, and $p(x,y)$ is the probability that the events x and y occur together.

The PMI is usually used to measure word similarity [11]. Consequently, it is utilized for the keyphrase and collocation extraction.

2.2. Natural Language Processing

NLP is a process that enables a computer to understand, analyze and process a natural language. Natural language is any language developed and used by human being such as English, Chinese, and so on. There are many researches studied in NLP, which include both text and speech processing [12][13]. In this sense, NLP is widely applied to data written by human on SNSs.

For such text processing, there are various useful researches in NLP such as the word segmentation, the part-of-speech tagging (POS tagging), and so on. The word segmentation is the process of dividing a written sentence into separate words. Basically, for languages such as English and Spanish, this task is very simple because words are mostly divided by a space. But, for languages like Chinese and Japanese, this task is a difficult problem because words are delimited. The POS tagging is the process of determining the part of speech of each word in a sentence. POS tagging algorithms are mostly divided into two approaches, statistic and rule-based algorithms. Besides, there are diverse processes for text processing such as named entity recognition, parsing, and morphological segmentation.

The Stanford Natural Language Processing Group has developed Natural Language Processing Software that provides us with various statistical, and rule-based NLP tools [14].

3. Methodology

3.1. Overview

In this section, we address a method to extract terms to be used as search keywords for crawling tweets. As mentioned in section 1, according to a topic, the topic term used as a single search keyword can cause collection of a lot of extraneous data in keyword-based search because of other different meanings. For this problem, data storage space can be wasted and the analysis of data becomes more difficult. Also, in SNS data collection in which resources to collect data such as search queries are limited, this problem cause inefficient use of the limited resources. Thus, for the topic in which the topic term is homonym, efforts to collect data more accurately in keyword-based search are needed. If using several terms relevant to a certain topic with the topic term as multiple search keywords, we can present the topic more obviously. Therefore, we can expect to collect data relevant with the topic more accurately.

It is important to select terms to be used with a topic term as multiple search keywords. The terms have to appear frequently in data relevant with the topic and appear as little as possible in data irrelevant with the topic. The terms selected according to these criteria can be helpful to collect relevant data more accurately.

In order to extract such terms for crawling tweets, first, we collect tweets using a topic term such as “coach” on keyword-based search, and then, we categorize the collected tweets into either relevant or irrelevant with hand-tagging task because the tweets collected by a topic term having different meanings basically include a lot of irrelevant tweets. After that, we transform the relevant tweets into a set of noun terms using Stanford NLP. We use only noun terms in order to extract terms for search

keywords. After this processing, we remove stopwords that mean terms that do not contain important meanings to be used as search keywords, finally, we calculate TF-IDF*PMI score of the terms to sort them in order of effective terms as search keywords. Figure 3 shows a method to extract terms to be used as search keywords for crawling tweets.

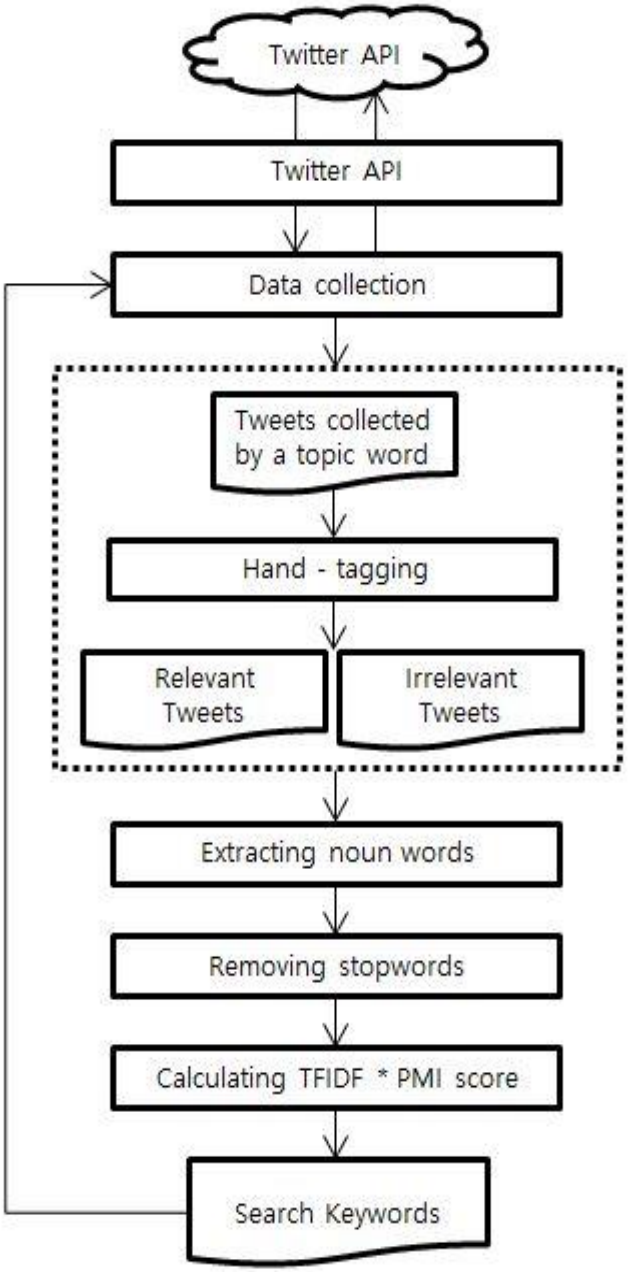

Figure 3. Overview of Method to Extract Search Keywords

3.2. Data Set for Extracting Search Keywords

It is important to construct a data set to be used to extract terms for search keywords. If the data set contains a lot of tweets relevant to a certain topic, we can extract a lot of terms relevant to the topic from the data set. In order to construct such data set, we can use three different ways with Twitter API. The first way is to categorize tweets into either relevant or irrelevant with hand-tagging task after collecting tweets using a topic term such as "coach" on keyword-based search. The collected tweets basically can include a lot of tweets irrelevant to the topic. Thus, to categorize the tweets is required in order to take tweets relevant to the topic. Table 1 is an example of categorized tweets with hand-tagging task.

Table 1. Example of Tweets Categorized with Hand-Tagging Task

Topic	Tweets	Categorization
Fashion Company "Coach"	Go enter to #win a beautiful stylish Coach Purse. @dinade #giveaway http://t.co/YakzaC2SM2	**Related**
	People still wear Coach shoes?	**Related**
	PU leather case for Plum Coach Plus II Z621 case cover - Only : US $12.75 USD http://t.co/HR9ZTzIOeW http://t.co/cZVbCwqy2P	Unrelated
	NFL News from ESPN Vermeil joins ownership of Arena League team http://t.co/zVVQkhSHPF	Unrelated
	This Coach private event thing with Ariana looks so good wtf	**Related**
Retail Company "Target"	#Facebook user creates fake Target customer support profile, trolls angry idiots - BGR http://t.co/4D5E5AF0G7	**Related**
	Nicolas Otamendi transfer latest: Manchester City target dropped by Valencia for Cham http://t.co/U6m00Atglw	Unrelated
	$OMER price target raised to $75 from $60 at WBB Securities	Unrelated
	New Mars Ice Cream Treats 6-Packs Coupon Means Only $1.54 At Target! via Couponing For 4 - The $1 ... http://t.co/CGoI8Sf0Tt	**Related**
	My cashier at Target looked like a young @KeriSpectrum	**Related**

The second way is to employ user's timeline search. Most of companies create an official Twitter ID, and share, and advertise their new products, events, and issues using the ID. Thus, if collecting tweets generated by an official Twitter ID created by a company selected as a topic, we can expect to extract terms related with the topic from these tweets. Table 2 is an example of tweets generated by an official Twitter ID.

Table 2. Example of Tweets Written on Timeline of Official Twitter ID

Writer	Tweets
Coach	Welcome to Instagram, Stuart Vevers! Follow along for a look at his #CoachFall2015 favorites, inspiration & more: https://t.co/VbsCbCwEsg
	Business style sharp enough to wear all week: #CoachMens2015 http://t.co/XRzzhOwXRV http://t.co/VeXLMTAPv7
	Congrats to our friends @streetdreamsnyc for being featured in the @nytimes! Great times together at #CoachMens2015. http://t.co/NftfL5bdE8
Target	And the winner of the snack bracket is...#Spicy! http://t.co/8mG9IyttRi
	Happy #StarWarsDay! Who's done this Jedi Door Trick? http://t.co/ZcajkoG8IT
	Hot styles full of fab savings. 15% off women's apparel with #Cartwheel. http://t.co/HGl7mbOpu7 http://t.co/2vV3hvnFVp

The last way is to collect tweets including a tag of an official Twitter ID relevant to a particular topic. Sometimes, Twitter users tag an official Twitter ID related with their stories while writing a tweet. Thus, tweets including a tag of an official Twitter ID relevant to a selected topic also can be expected to contain many relevant terms. We can collect such tweets using "@official Twitter ID" as a search keyword on keyword-based search. Table 3 shows an example of tweets including a tag of an official Twitter ID.

Table 3. Example of Tweets Including Tag of Official Twitter ID

Topic	Tweets
Fashion Company "Coach"	**@Coach** I have this Bandit tote and it's absolutely one of my favs!! Beautiful ❤
	@Coach Signature Small Wristlet for $27.99 (Reg. $48.00) + Free Shipping at @shop6pm! http://t.co/7Ui4alCnVn http://t.co/4O2b9OpgSV
Retail Company "Target"	Up to 60% OFF **@Target** ★ Up to 80% OFF @jcpenney #rcnocrop #cute #sale #jcpenney #target #wcw⋯ https://t.co/09QVnKm0Go
	If **@Target** clears their #Trophy shirt stock, I'll take 'em all! I've wanted to be objectified since I was 10 http://t.co/awwujKPZCQ

In order to evaluate terms extracted from the tweets collected with the three different ways explained above, we constructed three data sets using the collected tweets. Data set 1 is a set of tweets generated by an official Twitter ID related to a selected topic, data set 2 is a set of tweets including a tag of the official Twitter ID relevant to the topic, and data set 3 is a set of tweets categorized into relevant to the topic with hand-tagging task after collecting tweets using a topic term. We compared the terms extracted from the data sets with 50 terms we selected as terms that appears in the topic frequently. The 50 terms were selected based on Twitter search, and an official web site. Table 4 shows an example of the number of terms matched with the 50 terms and Table 5 shows an example of 50 terms we selected as terms that appears in a topic frequently.

Table 4. Number of Terms Matched with 50 Terms

Data set	**Number of Tweets**	**Number of matched terms**
Data set 1	150	14
Data set 2	150	19
Data set 3	150	44

Table 5. Selected 50 Terms Related with "Coach"

50 Terms Appearing Frequently in "coach"	bag, Japan, bags, mini, handbag, ebay, handbags, kelsey, f65536, celeste, NWT, phoebe, signature, carryall, crossbody, photos, black, photo, wallet, blue, gold, size, shoulder, khaki, hobo, sneakers, coin, sooyoung, f36181, ipad, event, clutches, purse, denim, wristlet, blush, satchel, heel, shoes, heels, box, canvas, tote, fashion, khaki, jewelry, giveaway, backpacks, ariana, accessory

The data set 3 (a set of tweets categorized into relevant with hand-tagging task after collecting tweets using a topic term) included the most terms matched with the 50 terms. The data set 1 and data set 2 have an advantage in that they don't need hand-tagging task unlike data set 3. So, we evaluated them with more tweets. Figure 4 shows an example of the number of terms matched with the 50 terms with data set 1 and 2.

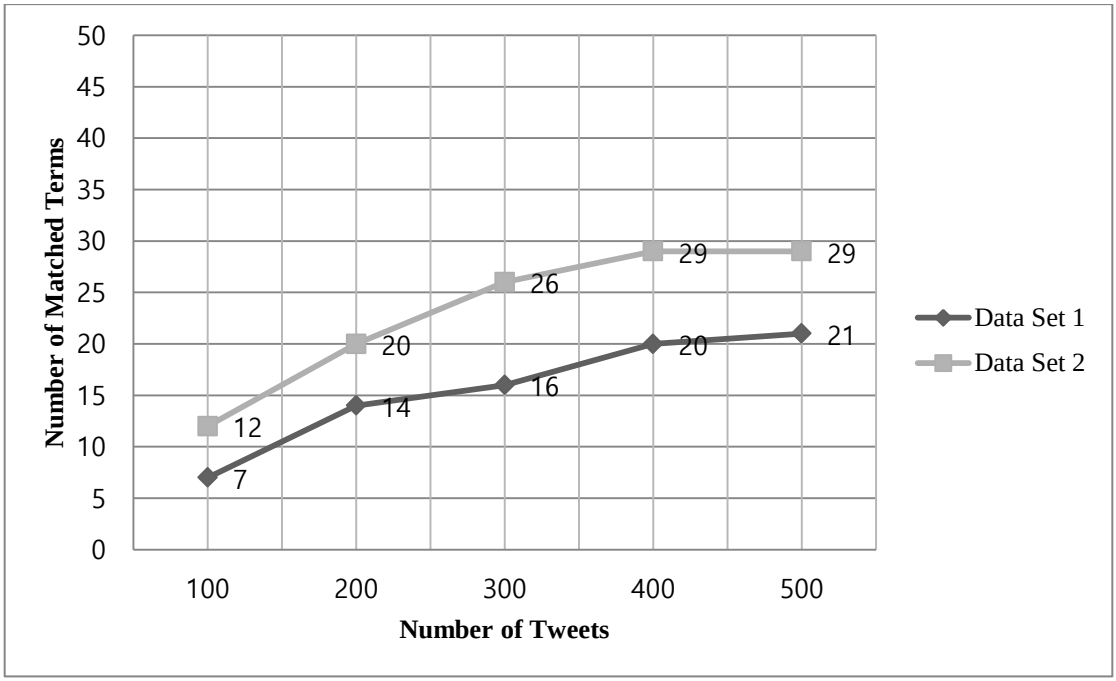

Figure 4. Number of Terms Matched with 50 Terms Using More Tweets

Although data set 1 and 2 were evaluated with more tweets than those of data set 3, the numbers of terms matched with the 50 term was very smaller. It means that we can

extract a lot of relevant terms from data set 3 compared with data set 1 and 2. Thus, we use data set 3 to extract terms for search keywords.

3.3. Term Extraction

After constructing a data set of relevant tweets to be used to extract terms for search keywords, we extract terms from the data set of relevant tweets using Stanford NLP. Stanford NLP provides useful analysis functions that find terms in a text, the base forms of the terms, and their parts of speech. Through Stanford NLP, the data set of relevant tweets is transformed into a set of terms. Also, we can extract terms with a specific part of speech using the Stanford NLP. We extract only noun terms because other parts of speech such as verb, adjective, and so on, can affect the analysis of data. For example, if a verb like “hate” is specified as a search keyword, we collect tweets including the term “hate”. In this case, analyzing the tweets will be distorted because the all collected tweets only have negative meaning. Table 6 shows an example of extracting terms from a data set of relevant tweets.

Table 6. Example of Set of Extracted Terms

Topic	Set of relevant tweets	Terms
Fashion Company “Coach”	I just entered to #win a COACH HANDBAG @poshonabudget #giveaway http://t.co/kATE6H0RDE Ariana at a private coach event in Japan. http://t.co/8A1Fd81haS	coach, handbag, poshonabudget, giveaway ariana, event, japan
Brand of Safety Razors “Gillette”	Buy me Gillette blades so I know it's real ?? @Cjstroz I found that out after I bought Gillette deodorant 1940's Gillette Gold Safety Razor (629) http://t.co/LNMUTNCCD8 http://t.co/97H92k7934	gillette, blade, cjstroz, deodorant, gold safety, razor

3.4. Stopword Removal

We remove stopwords from the terms extracted in the previous step. In this study, stopwords are terms not containing important meanings as search keywords. Firstly, terms containing uncertain meanings such as “-thing”, “-one”, and “-body” are regarded as stopwords. Secondly, topic terms such as “coach” and “target” are also considered as stopwords. We use topic terms as basic search keywords to collect tweets. Finally, term “rt” that means retweet, a way to republish a tweet that another Twitter user has written, is regarded as stopwords as well.

3.5. Search Keyword Selection

After stopword removal, we basically use TF-IDF and PMI to sort the terms in order of effective terms as search keywords. However, the terms sorted by TF-IDF score and PMI score are need to be more refined.

In the terms sorted by TF-IDF, some terms in high rank not only appear in relevant tweets frequently, but also appear in irrelevant tweets compared with the other terms in high rank. If using such terms as search keywords, both of relevant tweets and irrelevant tweets can be collected together. In the terms sorted by PMI, there is a sparse data problem [14]. Because of this problem, terms infrequently occurring in relevant tweets can be located in high rank.

Thus, we use TF-IDF*PMI score to minimize these problems. Using TF-IDF*PMI score, we can sort the term extracted from a set of relevant tweets in order of effective terms as search keywords better than when using TF-IDF score or PMI score. We extract the top 15% terms from 150 relevant tweets, which are sorted by TF-IDF*PMI score according experiment explained section 4.1.

3.5.1. TF-IDF

The basic concept of TF-IDF is proper to extract search keywords. The concept is to find terms frequently occurring in a particular document while not occurring in other documents in a corpus. Basically, a high TF-IDF score of a term is achieved by frequent occurrence of the term in the given document and infrequent occurrence in the other documents in a corpus. The higher TF-IDF score of a term is, the better the term represents the given document. In this sense, we use TF-IDF in order to find terms frequently occurring in a set of relevant tweets while not occurring in irrelevant tweets. As explained in section 3.2 to 3.5, we extract terms from a set of relevant tweets. If the terms are evaluated with high TF-IDF score, the term can represent the set of relevant tweets. Thus, they can be expected to be effective as search keywords because the terms occur frequently in the set of relevant tweets and occur infrequently in irrelevant tweets.

In this case, a bias towards longer document can occur because the set of relevant tweets is an exceedingly long document compared with irrelevant tweets. Thus, TF value normalized by the number of times all terms appears in a document is used to prevent the bias.TF of a term in a document can be defined as :

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Where $n_{i,j}$ is the number of times term i in document d_j and $\sum_k n_{k,j}$ is the number of time all terms appears in document d_j. Also, IDF of a term can be defined as :

$$idf_i = \log \frac{|D|}{|\{d_j : t_j \in d_j\}|}$$

Where $|\{d \in D : t \in d\}|$ is the number of documents where term *t* appears and N is total number of all documents in a corpus. Then, TF - IDF can be defined as :

$$tfidf_{i,j} = tf_{i,j} \text{ x } idf_i$$

For example, suppose that we have a corpus consisting of two irrelevant tweets and a set of relevant tweets on a topic “Gillette” as “Brand of safety razors”. First, we extract terms from the set of relevant tweets. After that, we evaluate importance of the term in the set of relevant tweets in the given corpus. Frequencies of the terms extracted from the set of relevant tweets in each document in the given corpus are shown in Table7.

Table 7. Example of Corpus

Document / Term	set of relevant tweets	irrelevant tweet 1	irrelevant tweet 2
blade	4	0	1
razor	3	0	0
safety	2	1	0

We can calculate TF-IDF score as explained above using the given corpus. Table 8 shows TF-IDF score of the terms in the set of relevant tweets.

Table 8. Example of TF-IDF

Term	TF	IDF	TF-IDF
blade	4 / 9 = 0.444	log(3 / 2) = 0.176	0.078
razor	3 / 9 = 0.333	log(3 / 1) = 0.477	0.159
safety	2 / 9 = 0.222	log(3 / 2) = 0.176	0.039

We calculate TF-IDF scores of all terms extracted a set of relevant tweets using corpus that consist irrelevant tweets and the set of relevant tweets. Table 9 shows the top 30 terms sorted by TF-IDF scores on a topic “Gillette” with a set of 150 relevant tweets and 849 irrelevant tweets.

Table 9. Top 30 Terms Sorted by TF-IDF Score in “Gillette”

Rank	Term	TF	IDF	TF-IDF
1	razor	95 / 966 = 0.098	log(850 / 1) = 2.929	0.288
2	fusion	37 / 966 = 0.038	log(850 / 1) = 2.929	0.112
3	proglide	28 / 966 = 0.028	log(850 / 1) = 2.929	0.084
4	richardsrazors	25 / 966 = 0.025	log(850 / 1) = 2.929	0.075
5	etsy	30 / 966 = 0.031	log(850 / 4) = 2.327	0.072
6	mach	23 / 966 = 0.023	log(850 / 1) = 2.929	0.069
7	blade	23 / 966 = 0.023	log(850 / 2) = 2.628	0.062
8	blades	15 / 966 = 0.015	log(850 / 1) = 2.929	0.045
9	handmade	14 / 966 = 0.014	log(850 / 1) = 2.929	0.042
10	kit	15 / 966 = 0.015	log(850 / 2) = 2.628	0.040
11	flexball	13 / 966 = 0.013	log(850 / 1) = 2.929	0.039
12	cartridges	12 / 966 = 0.012	log(850 / 1) = 2.929	0.036
13	safety	13 / 966 = 0.013	log(850 / 2) = 2.628	0.035
14	refills	11 / 966 = 0.011	log(850 / 1) = 2.929	0.033
15	mint	9 / 966 = 0.009	log(850 / 1) = 2.929	0.027
16	rs	9 / 966 = 0.009	log(850 / 1) = 2.929	0.027
17	power	10 / 966 = 0.010	log(850 / 2) = 2.628	0.027
18	count	8 / 966 = 0.008	log(850 / 1) = 2.929	0.024
19	amazon	8 / 966 = 0.008	log(850 / 2) = 2.628	0.021
20	fat	7 / 966 = 0.007	log(850 / 1) = 2.929	0.021
21	1930s	7 / 966 = 0.007	log(850 / 1) = 2.929	0.021
22	manual	7 / 966 = 0.007	log(850 / 1) = 2.929	0.021
23	ebay	10 / 966 = 0.010	log(850 / 8) = 2.026	0.020
24	new	13 / 966 = 0.013	log(850 / 27) = 1.498	0.020
25	boy	9 / 966 = 0.009	log(850 / 8) = 2.026	0.018
26	razors	6 / 966 = 0.006	log(850 / 1) = 2.929	0.018
27	adjustable	6 / 966 = 0.006	log(850 / 1) = 2.929	0.018
28	tech	6 / 966 = 0.006	log(850 / 2) = 2.628	0.016
29	cartridge	5 / 966 = 0.005	log(850 / 1) = 2.929	0.015
30	men	5 / 966 = 0.005	log(850 / 2) = 2.628	0.013

3.5.2 PMI

PMI is a measure of association between particular events x and y. In this thesis, we use PMI to measure association between occurrence of a topic term as a selected topic such as “Gillette” meaning as “Brand of safety razors” and occurrences of terms extracted from a set of relevant tweets such as “razor”, ”safety”, and “fusion”. In this case, a high PMI score means a term extracted from a set of relevant tweets occurs only with a topic term meaning as a selected topic. Thus, a term with a high PMI score is strongly related with the topic. And consequently, it can be expected to be effective as search keywords. PMI can be defined as :

$$pmi(x, y) = \log \frac{p(x,\ y)}{p(x)p(y)}$$

Where $p(x)$ is the probability that the topic term x meant as a particular topic occurs in all tweets collected by the topic term, $p(y)$ is the probability that the term y extracted from a set of relevant tweets occurs in all the collected tweets, and $p(x, y)$ is a probability that the topic term x meant as the particular topic and the term y extracted from the set of relevant tweets occurring together in all the collected tweets.

For example, suppose that we have the categorized tweets on a topic “Gillette” as “Brand of safety razors”. The categorized tweets consist of 4 relevant tweets and 6 irrelevant tweets. We calculate PMI score of a term “razor” extracted from a set of the relevant tweets. In this case, we can assume that the topic term occurring in relevant tweets is meant as the selected topic and the topic term occurring in irrelevant tweets is meant as the different meanings. Therefore, for $p(x)$, we count occurrence of the term “Gillette” in the relevant tweets. Also, for $p(y)$, we count occurrence of the term “razor” in the all tweets. Finally, for $p(x, y)$, we count occurrence of the term “razor”

with “Gillette” in relevant tweets. Now, we can calculate PMI score of the term “razor” with the topic term “Gillette” as “Brand of safety razors” as follows:

$$p(gillette) = \frac{4}{10}$$

$$p(razor) = \frac{6}{10}$$

$$p(gillette, razor) = \frac{4}{10}$$

$$pmi(gillette, razor) = \log \frac{\frac{4}{10}}{\frac{4}{10} \text{ X } \frac{6}{10}} = 0.2218$$

We calculate PMI scores of all terms extracted a set of relevant tweets with irrelevant tweets. Table 10 shows the top 30 terms sorted by PMI scores on a topic “Gillette” with a set of 150 relevant tweets and 849 irrelevant tweets.

Table 10. Top 30 Terms Sorted by PMI Score in "Gillette"

Rank	Term	$p(x)$	$p(y)$	$p(x,y)$	$pmi(x,y)$
1	razor	150 / 999	80 / 999	80 / 999	0.823
2	fusion	150 / 999	36 / 999	36 / 999	0.823
3	proglide	150 / 999	27 / 999	27 / 999	0.823
4	richardsrazors	150 / 999	25 / 999	25 / 999	0.823
5	blades	150 / 999	14 / 999	14 / 999	0.823
6	flexball	150 / 999	13 / 999	13 / 999	0.823
7	handmade	150 / 999	12 / 999	12 / 999	0.823
8	mach	150 / 990	12 / 999	12 / 999	0.823
9	cartridges	150 / 999	12 / 999	12 / 999	0.823
10	refills	150 / 999	11 / 999	11 / 999	0.823
11	mint	150 / 999	9 / 999	9 / 999	0.823
12	rs	150 / 999	9 / 999	9 / 999	0.823
13	count	150 / 999	8 / 999	8 / 999	0.823
14	fat	150 / 999	7 / 999	7 / 999	0.823
15	1930s	150 / 999	7 / 999	7 / 999	0.823
16	manual	150 / 999	7 / 999	7 / 999	0.823
17	razors	150 / 999	6 / 999	6 / 999	0.823
18	adjustable	150 / 999	6 / 999	6 / 999	0.823
19	cartridge	150 / 999	5 / 999	5 / 999	0.823
20	ebony	150 / 999	4 / 999	4 / 999	0.823
21	refill	150 / 999	4 / 999	4 / 999	0.823
22	mach3	150 / 999	3 / 999	3 / 999	0.823
23	crafts	150 / 999	3 / 999	3 / 999	0.823
24	standard	150 / 999	3 / 999	3 / 999	0.823
25	fatboy	150 / 999	3 / 999	3 / 999	0.823
26	brush	150 / 999	3 / 999	3 / 999	0.823
27	cream	150 / 999	3 / 999	3 / 999	0.823
28	apitconnect	150 / 999	3 / 999	3 / 999	0.823
29	slim	150 / 999	3 / 999	3 / 999	0.823
30	hipsters	150 / 999	2 / 999	2 / 999	0.823

3.5.3 TF-IDF*PMI

Terms to be used as search keywords must be selected carefully. The Terms have to appear in relevant data frequently and appear in irrelevant data as little as possible. We use TF-IDF and PMI to select such terms. However, there are some problems with the two statistic methods.

In the terms sorted by TF-IDF score presented in table 10, some terms occurring in relevant tweets and irrelevant tweets together are in high rank. For example, “ebay”, “new” and “boy” frequently appeared in relevant tweets, also, appeared in irrelevant tweets compared with the other terms in the high rank. If such terms are used as search keywords, collection of both relevant and irrelevant tweets can be caused. Thus, the other terms in the high rank that infrequently occurred in irrelevant tweets must be in higher rank.

Also, in the terms sorted by PMI score given in table 11, Because of the sparse data problem, Terms occurring in relevant tweets such as “apitconnect”,”slim”, and “hipsters” were located in high rank than terms very frequently occurring in relevant tweets such as “esty”, “blade”, and “kit”. We cannot expect the terms such as “apitconnect”,”slim”, and “hipsters” are more effective for search keywords than the terms such as “esty”, “blade”, and “kit”.

Thus, to mitigate these problems in TF-IDF and PMI, we consider the score of TF-IDF multiplied by PMI. Table 11 shows 30 terms sorted by the score of TF-IDF multiplied by PMI with the same tweets used in Table 9 and Table 10. By considering the score of TF-IDF multiplied by PMI, we can extract the rank of terms rationally.

Table 11. Top 30 Terms Sorted by TF-IDF*PMI Score in "Gillette"

Rank	Term	TF-IDF	PMI	TF-IDF * PMI
1	razor	0.288	0.823	0.237
2	fusion	0.112	0.823	0.092
3	proglide	0.084	0.823	0.069
4	richardsrazors	0.075	0.823	0.062
5	mach	0.069	0.823	0.057
6	etsy	0.072	0.780	0.056
7	blade	0.062	0.803	0.050
8	blades	0.045	0.823	0.037
9	handmade	0.042	0.823	0.034
10	kit	0.040	0.795	0.032
11	flexball	0.039	0.823	0.032
12	cartridges	0.036	0.823	0.029
13	safety	0.035	0.791	0.027
14	refills	0.033	0.823	0.027
15	mint	0.027	0.823	0.022
16	rs	0.027	0.823	0.022
17	power	0.027	0.777	0.021
18	count	0.024	0.823	0.019
19	fat	0.021	0.823	0.017
20	1930s	0.021	0.823	0.017
21	manual	0.021	0.823	0.017
22	amazon	0.021	0.772	0.016
23	razors	0.018	0.823	0.014
24	adjustable	0.018	0.823	0.014
25	cartridge	0.015	0.823	0.012
26	ebay	0.020	0.593	0.012
27	tech	0.016	0.756	0.012
28	boy	0.018	0.573	0.010
29	men	0.013	0.744	0.010
30	ebony	0.012	0.823	0.009

To compare the three different ways to sort terms in order of effective terms as search keywords, TF-IDF score, PMI score, and TF-IDF*PMI score, we counted the frequencies of the terms in 150 relevant tweets and 849 irrelevant tweets used to sort the terms. Figure 5 shows the frequencies of terms sorted by TF-IDF, PMI, and TF-IDF*PMI scores in 150 relevant tweets.

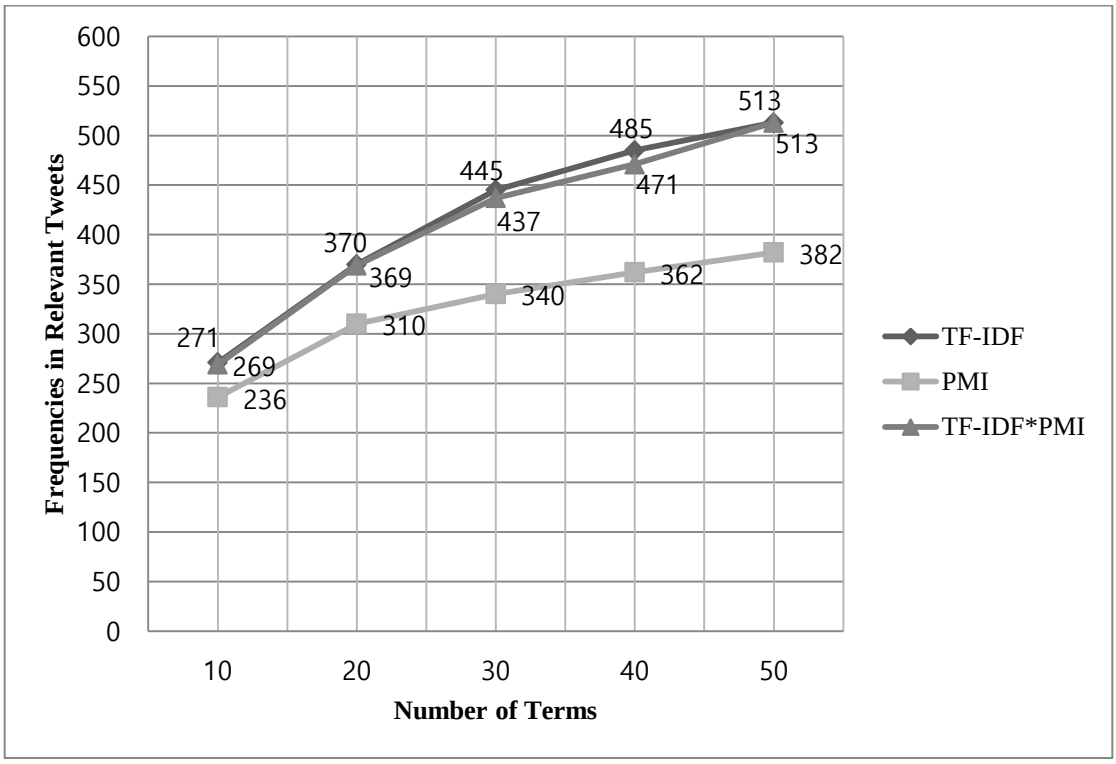

Figure 5. Frequencies of Terms Sorted by TF-IDF, PMI, TF-IDF*PMI Score in Relevant Tweets

The terms sorted by TF-IDF scores and TF-IDF*PMI scores appeared in the relevant tweets more frequently than the terms sorted by PMI scores. Thus, we can assume that the terms sorted by TF-IDF scores and TF-IDF*PMI scores can be more effective to be used as search keywords than the terms sorted by PMI scores.

Also, the frequencies of the terms sorted by TF-IDF, PMI, and TF-IDF*PMI in the 849 irrelevant tweets are shown in Figure 6.

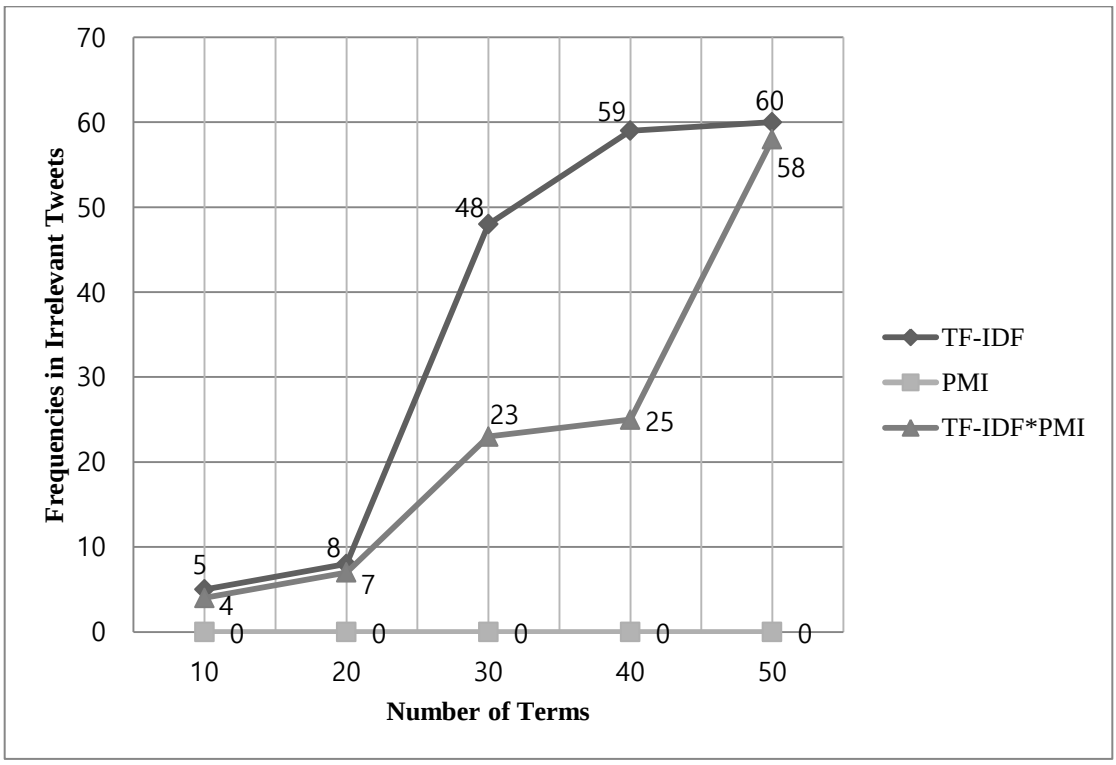

Figure 6. Frequencies of Terms Sorted by TF-IDF, PMI, TF-IDF*PMI Score in Irrelevant Tweets

The terms sorted by TF-IDF scores appeared in the irrelevant tweets more frequently than the terms sorted by PMI scores or TF-IDF*PMI scores. Also the terms sorted by PMI score most infrequently appeared in the irrelevant tweets, but, as shown in figure 5, the terms appeared in the relevant tweets most infrequently. Thus, we can assume that the terms sorted by TF-IDF*PMI scores are the most effective terms as search keywords.

4. Experiment Result

In this section, first, we explain an experiment to determine thresholds for proper number of relevant tweets to extract terms and number of the terms extracted by the proposed method for crawling tweets. And then, we discuss a result of tweet collection using the threshold values.

We selected four topics in which a lot of extraneous tweets occur when collecting tweets using a topic term such as "coach" as a single search keyword. We randomly collected 100 tweets on the four topics 10 times without duplication. The averages of the number of relevant tweets and irrelevant tweets on the four topics are shown in table 12.

Table 12. Topics

Topic	Category	Search Keyword	Relevant Tweets	Irrelevant Tweets
Coach	Fashion company	"Coach"	11	89
Gillette	Brand of safety razors	"Gillette"	15	85
Sonata	Car	"Sonata"	31	69
Target	Retailing Company	"Target"	36	64

We utilized the F-measure to evaluate all experiments explained above. The F-measure is a measure that combines precision and recall. The Precision and recall is widely used to measure the performance of information retrieval [15].

The precision is the probability that retrieved documents are relevant and the recall is the probability that relevant documents are retrieved. The measures are defined as:

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

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

4.1. Experiment for Thresholds

To evaluate various thresholds for the proper number of relevant tweets to extract terms and number of the terms to be used as multiple search keywords based on the F-measure score, we implemented a search system that retrieve tweets stored in our database. Real tweet collection using keyword-based search on Twitter API, we can collect recent tweets from the last 6 - 9 days [16]. Thus, it is a difficult task to evaluate the F-measure for various thresholds with the same tweets based on real tweet collection. The tweets to be retrieved in our system consisted of 1000 tweets on the each topic. 1000 tweets were collected using a topic term for each topic. Table 13 shows 1000 tweets to be retrieved on the each topic.

Table 13. 1000 Tweets to Be Searched in Our Search System

Topic	N of Relevant Tweets	N of Irrelevant Tweets	Total tweets
Coach	134	866	1,000
Gillette	152	848	1,000
Sonata	224	776	1,000
Target	209	791	1,000

Also, we constructed from 50 to 200 relevant tweets to extract terms to be used as multiple search keywords. Also, we used irrelevant tweets to calculate TF-IDF*PMI. The number of relevant tweets and irrelevant tweets are given in Table 14. Also these tweets don’t include any of tweets to be retrieved shown in Table 13. Table 14 shows

Table 14. 50 to 200 Relevant Tweets to Extract Search Keywords

Topic	Relevant tweets	Irrelevant tweets	Total tweets
Coach	50	404	454
	100	796	896
	150	1,178	1,328
	200	1,543	1,743
Gillette	50	283	333
	100	544	644
	150	849	999
	200	1,156	1,356
Sonata	50	132	182
	100	223	323
	150	324	474
	200	487	687
Target	50	92	142
	100	176	276
	150	289	439
	200	375	575

We extracted terms from relevant tweets given in Table 14, and then, we calculated TF-IDF*PMI scores of the terms. Using the top N% terms, we searched the tweets given in Table 13 using the search system we implemented.

To find the proper number of relevant tweets to extract terms, we used maximal F-score measured from 50 to 200 relevant tweets on the each topic. For example, when we used 150 relevant tweets, 15% terms showed the maximal F-score in a topic “Sonata”. As the result, the average of maximal f-measure score is on 150 and 200 relevant tweets. Thus, we choose 150 relevant tweets as threshold for proper number of relevant tweets to extract terms. Table 15 and figure 7 shows the averages of maximal f- measure score.

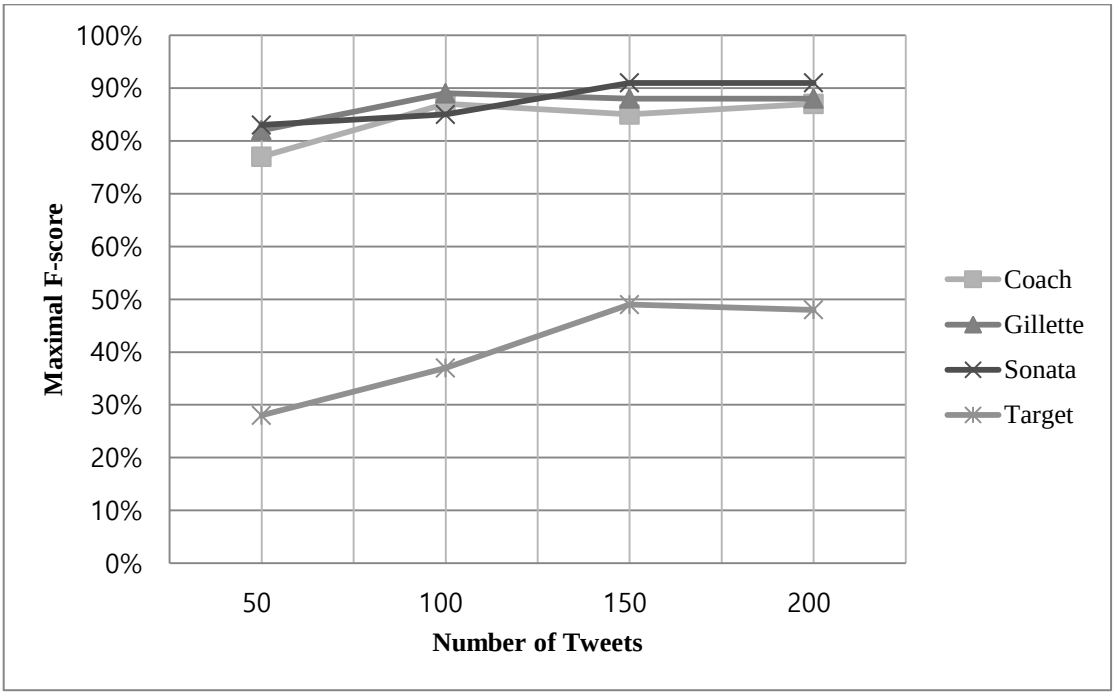

Figure 7. Maximal F-measure Scores in 50 to 200 Relevant Tweets

Table 15. Averages of Maximal F-measure Scores

N of Relevant Tweets	Coach	Gillette	Sonata	Target	Average
50	0.77	0.82	0.83	0.28	0.67
100	0.87	0.89	0.85	0.37	0.74
150	0.85	0.88	0.91	0.49	0.78
200	0.87	0.88	0.91	0.48	0.78

Also, to determine the number of the terms to be used as multiple search keywords, we considered averages of F-measure scores when using up to 50% terms with the 150 relevant tweets for all topics. The highest average of F-measure scores is on the top 15%. Thus, we determined the top 15% terms extracted from the relevant tweets as multiple search keywords. Table 16 and figure 8 show the F-measure scores with 150 relevant tweets for all topics. Also figures 9 to 12 and tables 17 to 20 show the f-measure scores on 50% terms extracted from 50 to 200 relevant tweets for each topic.

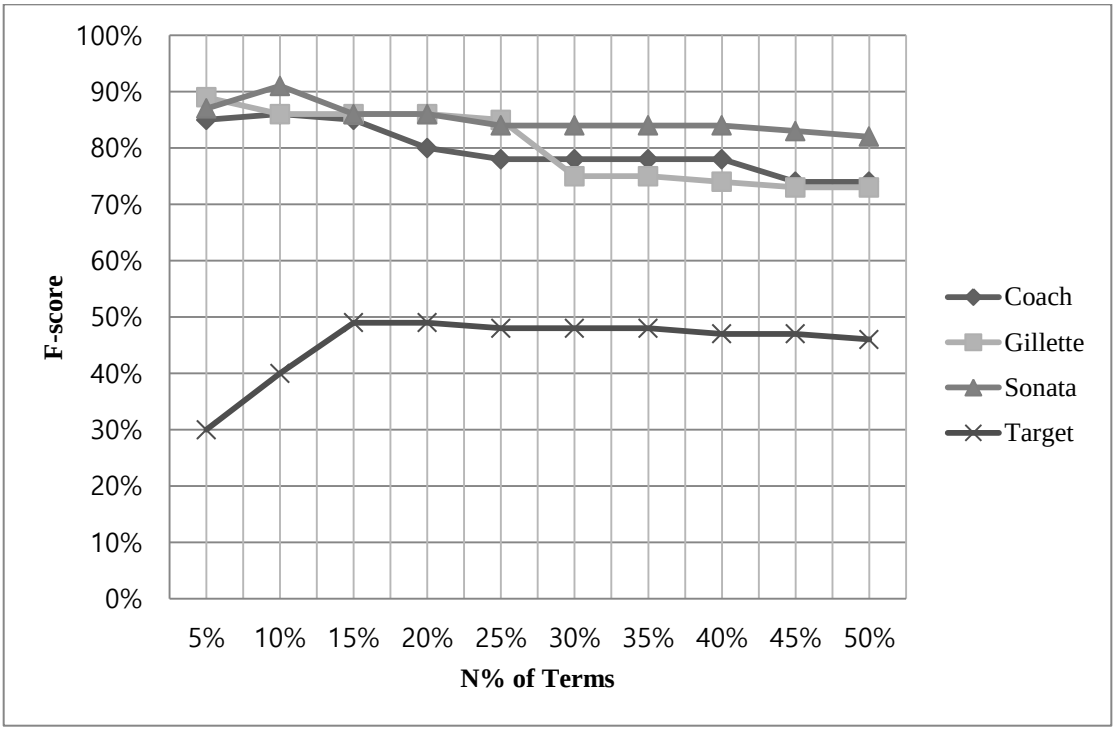

Figure 8. F-measure Score with 150 Relevant Tweets

Table 16. Averages of F-measure Scores with 150 Relevant Tweets

N% of Term	Coach	Gillette	Sonata	Target	Average
5%	0.85 (0.91/0.79)	0.88 (0.91/0.86)	0.87 (0.96/0.8)	0.30 (0.52/0.21)	0.72
10%	0.86 (0.86/0.85)	0.86 (0.83/0.89)	0.91 (0.95/0.87)	0.40 (0.52/0.32)	0.75
15%	0.85 (0.85/0.86)	0.86 (0.81/0.92)	0.86 (0.83/0.90)	0.49 (0.53/0.46)	0.76
20%	0.80 (0.68/0.96)	0.86 (0.80/0.92)	0.86 (0.82/0.91)	0.49 (0.52/0.47)	0.75
25%	0.78 (0.65/0.97)	0.85 (0.79/0.92)	0.84 (0.78/0.92)	0.48 (0.49/0.48)	0.73
30%	0.79 (0.66/0.98)	0.75 (0.63/0.93)	0.84 (0.77/0.92)	0.48 (0.48/0.49)	0.71
35%	0.78 (0.65/0.99)	0.75 (0.63/0.94)	0.84 (0.77/0.92)	0.48 (0.46/0.50)	0.71
40%	0.78 (0.65/0.99)	0.74 (0.61/0.94)	0.84 (0.77/0.92)	0.47 (0.44/0.50)	0.70
45%	0.74 (0.59/0.99)	0.73 (0.60/0.94)	0.83 (0.75/0.93)	0.47 (0.44/0.51)	0.69
50%	0.74 (0.59/0.99)	0.73 (0.59/0.94)	0.82 (0.74/0.93)	0.46 (0.41/0.52)	0.68

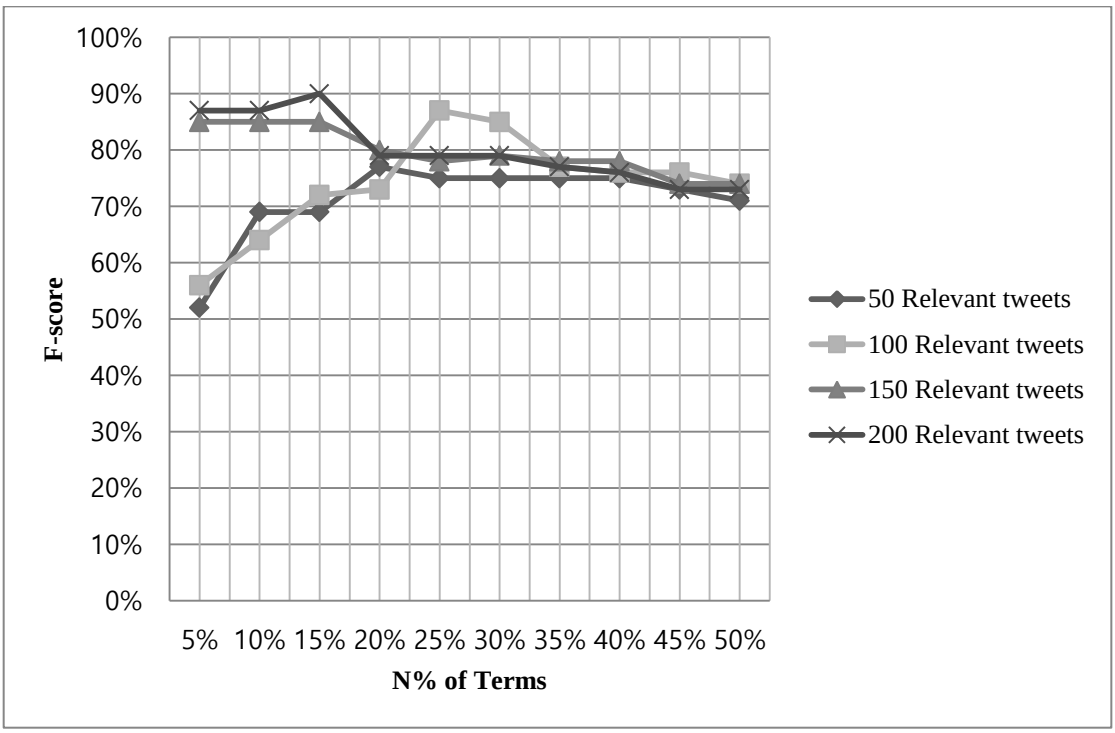

Figure 9. F-measure Scores in 50 to 200 Relevant Tweets in "Coach"

Table 17. F-measure Scores in 50 to 200 Relevant Tweets in "Coach"

N% of Term	50	100	150	200
5%	0.52 (0.96/0.36)	0.56 (0.96/0.40)	0.85 (0.91/0.79)	0.87 (0.92/0.83)
10%	0.69 (0.93/0.55)	0.64 (0.88/0.50)	0.85 (0.86/0.85)	0.87 (0.91/0.84)
15%	0.69 (0.91/0.56)	0.72 (0.83/0.63)	0.85 (0.85/0.86)	0.87 (0.83/0.92)
20%	0.77 (0.92/0.66)	0.73 (0.83/0.65)	0.80 (0.68/0.96)	0.79 (0.66/0.97)
25%	0.75 (0.83/0.68)	0.87 (0.85/0.90)	0.78 (0.65/0.97)	0.79 (0.66/0.98)
30%	0.75 (0.83/0.68)	0.85 (0.80/0.90)	0.79 (0.66/0.98)	0.79 (0.66/0.98)
35%	0.75 (0.83/0.68)	0.77 (0.64/0.96)	0.78 (0.65/0.99)	0.77 (0.63/0.99)
40%	0.75 (0.83/0.69)	0.76 (0.63/0.96)	0.78 (0.65/0.99)	0.76 (0.62/0.99)
45%	0.73 (0.65/0.82)	0.76 (0.62/0.98)	0.74 (0.59/0.99)	0.73 (0.58/0.99)
50%	0.71 (0.63/0.82)	0.74 (0.59/0.98)	0.74 (0.59/0.99)	0.73 (0.58/0.99)

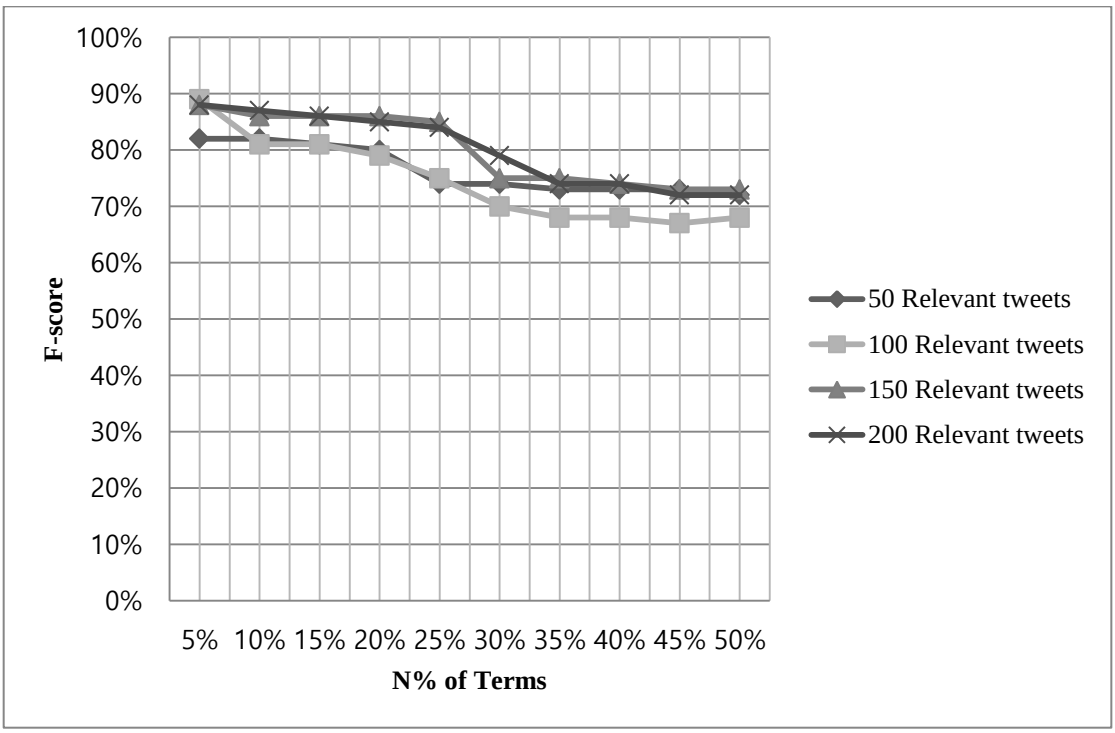

Figure 10. F-measure Scores in 50 to 200 Relevant Tweets in "Gillette"

Table 18. F-measure Scores in 50 to 200 Relevant Tweets in "Gillette"

N% of Term	50	100	150	200
5%	0.82 (0.79/0.86)	0.89 (0.92/0.86)	0.88 (0.91/0.86)	0.88 (0.87/0.89)
10%	0.82 (0.76/0.89)	0.81 (0.75/0.89)	0.86 (0.83/0.89)	0.87 (0.83/0.91)
15%	0.81 (0.74/0.89)	0.81 (0.73/0.90)	0.86 (0.81/0.92)	0.86 (0.81/0.91)
20%	0.80 (0.72/0.91)	0.79 (0.71/0.90)	0.86 (0.80/0.92)	0.85 (0.79/0.92)
25%	0.74 (0.61/0.95)	0.75 (0.62/0.95)	0.85 (0.79/0.92)	0.84 (0.78/0.92)
30%	0.74 (0.60/0.97)	0.70 (0.55/0.95)	0.75 (0.63/0.93)	0.79 (0.68/0.93)
35%	0.73 (0.59/0.97)	0.68 (0.53/0.95)	0.75 (0.63/0.94)	0.74 (0.61/0.94)
40%	0.73 (0.59/0.97)	0.68 (0.53/0.95)	0.74 (0.61/0.94)	0.74 (0.61/0.94)
45%	0.73 (0.58/0.97)	0.67 (0.52/0.95)	0.73 (0.60/0.94)	0.72 (0.59/0.94)
50%	0.72 (0.57/0.97)	0.68 (0.52/0.97)	0.73 (0.59/0.94)	0.72 (0.59/0.94)

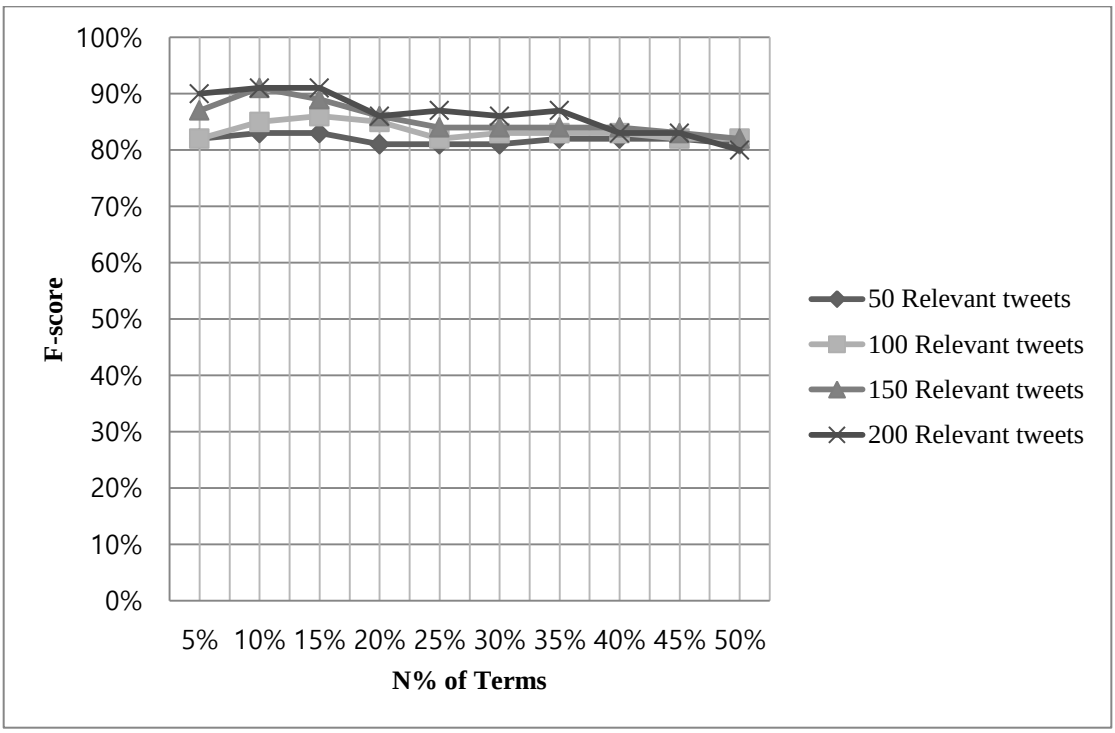

Figure 11. F-measure Scores in 50 to 200 Relevant Tweets in "Sonata"

Table 19. F-measure Scores in 50 to 200 Relevant Tweets in "Sonata"

N% of Term	50	100	150	200
5%	0.82 (1.0/0.70)	0.82 (1.0/0.70)	0.87 (0.96/0.8)	0.90 (0.96/0.84)
10%	0.83 (0.86/0.79)	0.85 (0.83/0.87)	0.91 (0.95/0.87)	0.91 (0.93/0.89)
15%	0.80 (0.84/0.82)	0.85 (0.83/0.88)	0.86 (0.83/0.90)	0.91 (0.93/0.90)
20%	0.81 (0.79/0.82)	0.85 (0.81/0.89)	0.86 (0.82/0.91)	0.87 (0.82/0.92)
25%	0.81 (0.79/0.83)	0.81 (0.75/0.89)	0.84 (0.78/0.92)	0.87 (0.81/0.93)
30%	0.82 (0.79/0.83)	0.83 (0.76/0.91)	0.84 (0.77/0.92)	0.87 (0.81/0.93)
35%	0.82 (0.78/0.86)	0.83 (0.76/0.92)	0.84 (0.77/0.92)	0.87 (0.81/0.93)
40%	0.82 (0.78/0.86)	0.83 (0.75/0.92)	0.84 (0.77/0.92)	0.83 (0.75/0.93)
45%	0.82 (0.78/0.86)	0.83 (0.75/0.92)	0.83 (0.75/0.93)	0.83 (0.75/0.93)
50%	0.81 (0.77/0.86)	0.82 (0.75/0.92)	0.82 (0.74/0.93)	0.81 (0.71/0.93)

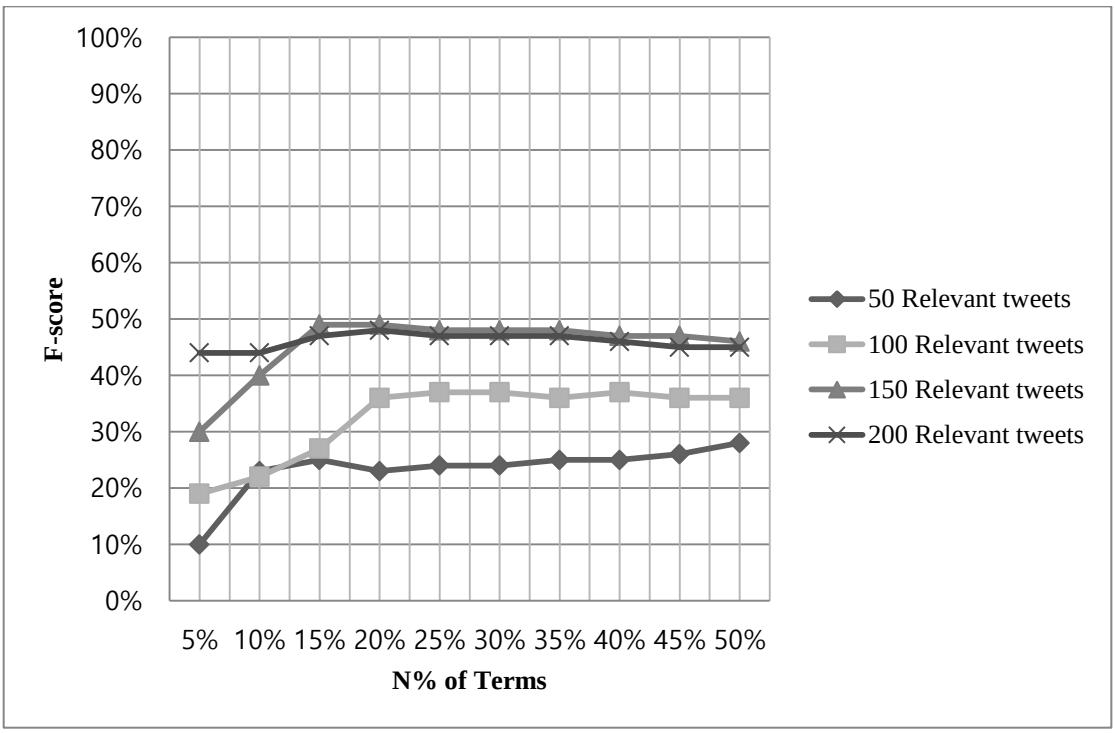

Figure 12. F-measure Scores in 50 to 200 Relevant Tweets in "Target"

Table 20. F-measure Scores in 50 to 200 Relevant Tweets in "Target"

N% of Term	50	100	150	200
5%	0.10 (0.36/0.06)	0.19 (0.47/0.12)	0.30 (0.52/0.21)	0.44 (0.62/0.34)
10%	0.23 (0.42/0.16)	0.22 (0.36/0.16)	0.40 (0.52/0.32)	0.44 (0.55/0.37)
15%	0.25 (0.42/0.18)	0.27 (0.36/0.21)	0.49 (0.53/0.46)	0.47 (0.54/0.41)
20%	0.23 (0.31/0.18)	0.36 (0.44/0.31)	0.49 (0.52/0.47)	0.48 (0.49/0.47)
25%	0.24 (0.31/0.20)	0.37 (0.43/0.32)	0.48 (0.49/0.48)	0.47 (0.47/0.48)
30%	0.24 (0.31/0.20)	0.37 (0.41/0.33)	0.48 (0.48/0.49)	0.47 (0.45/0.49)
35%	0.25 (0.31/0.21)	0.36 (0.38/0.35)	0.48 (0.46/0.50)	0.47 (0.43/0.51)
40%	0.25 (0.31/0.21)	0.37 (0.38/0.36)	0.47 (0.44/0.50)	0.46 (0.42/0.52)
45%	0.26 (0.28/0.25)	0.36 (0.37/0.36)	0.47 (0.44/0.51)	0.45 (0.40/0.52)
50%	0.28 (0.29/0.28)	0.36 (0.36/0.37)	0.46 (0.41/0.52)	0.45 (0.39/0.54)

4.2. Experiment for Tweet Collection with Our Method

In this section, we present the result of tweet collection using the thresholds presented in section 4.1. We compared tweet collection using a topic term as a single search keyword and tweet collection using a topic term with terms extracted using the proposed method as multiple search keywords. To evaluate the performances of two tweet collection, we utilized the precision, recall and F-measure for each topic.

4.2.1. Data Set

We used top 15% terms 150 relevant tweets and irrelevant tweets shown in table 21 to extract terms to be used as search keywords. The top 15% terms is also shown in table 22.

Table 21. Data Set to Extract Search Keywords

Topic	Number of Relevant Tweets	Number of Irrelevant Tweets	Total tweets
Coach	150	1,178	1,328
Gillette	150	849	999
Sonata	150	324	474
Target	150	289	439

Table 22. Number of Top 15% Term

Topic	Number of 15% Terms
Coach	28
Gillette	53
Sonata	49
Target	70

4.2.2. Result

To evaluate the results of tweet collection using a topic term as a single search keyword and tweet collection using a topic term with 15% extracted terms as multiple search keywords, we sampled the collected tweets shown in table 23.

Table 23. Tweet Generation Time of Sampled Tweet

Topic	Sample Tweet Generation Time
Coach	09. 11. 2015 19 : 30 ~ 20 : 00
Gillette	09. 11. 2015 14 : 00 ~ 20 : 00
Sonata	08. 31. 2015 00 : 00 ~ 23 : 59
Target	09. 31. 2015 19 : 30 ~ 20 : 00

We could collect a small amount of tweets using a topic term and 15% terms as multiple search keywords than when using a topic term as a single search keyword. The number of sampled tweets collected by a topic term as a single search keyword and the topic term with 15% extracted terms as multiple search keywords is shown in table 24.

Table 24. Number of Sampled Tweets

Topic	Search Keyword	Number of Collected Tweets
Coach	“Coach”	1,131
	“Coach” and 28 terms	48
Gillette	“Gillette”	902
	“Gillette” and 53 terms	68
Sonata	“Sonata”	792
	“Sonata” and 49 terms	269
Target	“Target”	1,098
	“Target” and 70 terms	206

First, the performance of tweet collection using a topic term with 15% extracted term is shown in Table 25.

Table 25. F-measure of Tweet Collection Using Topic Term and Top 15% Terms

Search Keyword	Precision	Recall	F-measure
“Coach” and 28 terms	0.75 (36 / 48)	0.69 (36 / 52)	0.71
“Gillette” and 53 terms	0.70 (48 / 68)	0.64 (48 / 75)	0.66
“Sonata” and 49 terms	0.96 (260 / 269)	0.94 (260 / 276)	0.94
“Target” and 70 terms	0.65 (134 / 206)	0.37 (134 / 361)	0.47

Table 26 illustrates the precision of tweet collection using a topic term as single keywords and using the topic term and 15% terms as multiple keywords. As the result, we could collect tweets very accurately compared tweet collection using a topic term as a single search keyword in all topics we addressed.

Table 26. Comparison of Two Tweet Collection on Precision

Topic	Search Keyword	Precision
Coach	“Coach”	0.04 (52 / 1,131)
	“Coach” and 28 terms	0.75 (36 / 48)
Gillette	“Gillette”	0.08 (75 / 902)
	“Gillette” and 53 terms	0.70 (48 / 68)
Sonata	“Sonata”	0.34 (276 / 792)
	“Sonata” and 49 terms	0.96 (260 / 269)
Target	“Target”	0.32 (361 / 1,098)
	“Target” and 70 terms	0.65 (134 / 206)

4.2.3. Result (Recursive)

SNS users are very sensitive to new products, trends, and events. Thus, search keywords for crawling tweets are needed to be updated. We extracted new search keywords from 150 tweets collected by the initial search keywords. Also, to calculate TF-IDF*PMI, we used the same irrelevant tweets shown in table 21. Table 27 illustrates the number of 15% terms on the each topic for new search keywords.

Table 27. Number of Top 15% Term

Topic	N of 15% terms
Coach	32
Gillette	58
Sonata	42
Target	75

Also, in order to measure the performance of tweet collection using new search keywords, we sampled tweets shown in Table 28.

Table 28. Sampled Tweet

Topic	Sample Tweet Generation Time
Coach	09. 13. 2015 19 : 30 ~ 20 : 00
Gillette	09. 16. 2015 14 : 00 ~ 20 : 00
Sonata	09. 16. 2015 19 : 30 ~ 19 : 59
Target	09. 16. 2015 00 : 00 ~ 23 : 59

We could collect a small amount of tweets using a topic term and new 15% terms as multiple search keywords. The number of sampled tweets collected by a topic term

as a single search keyword and the topic term with new 15% extracted terms as multiple search keywords is shown in table 29.

Table 29. Ratio of Data Related to Topic in Twitter

Topic	Search Keyword	Number of Collected Tweets
Coach	"Coach"	1,348
	"Coach" and 32 terms	77
Gillette	"Gillette"	834
	"Gillette" and 58 terms	104
Sonata	"Sonata"	737
	"Sonata" and 42 terms	194
Target	"Target"	873
	"Target" and 75 terms	228

The performance of tweet collection using a topic term with new 15% extracted term is shown in Table 30. As the result, in cases of "Coach" and "Gillette", F-measure scores were increased compared tweet collection using the topic term and initial 15% terms as multiple keywords. In case of "Sonata", F-measure score was decreased slightly. In case of "Target", F-measure score was hold.

Table 30. F-measure of Tweet Collection Using Topic Term and Top 15% Terms

Search Keyword	Precision	Recall	F-measure
"Coach" and 32 terms	0.90 (70/77)	0.93 (70/75)	0.91
"Gillette" and 58 terms	0.83 (87/104)	0.83 (87/107)	0.83
"Sonata" and 42 terms	0.96 (181/194)	0.91 (181/198)	0.91
"Target" and 75 terms	0.62 (142/228)	0.39 (142/361)	0.47

Table 31 illustrates the precision of tweet collection using a topic term as a single keyword and using the topic term and new 15% terms as multiple keywords. As the result, we could collect tweets very accurately as the first tweet collection.

Table 31. Comparison of Two Tweet Collection on Precision

Topic	Search Keyword	Precision
Coach	"Coach"	0.05 (75/1,348)
	"Coach" and 32 terms	0.90 (70/77)
Gillette	"Gillette"	0.12 (107/834)
	"Gillette" and 58 terms	0.83 (87/104)
Sonata	"Sonata"	0.26 (198/737)
	"Sonata" and 42 terms	0.96 (181/194)
Target	"Target"	0.32 (361/1,098)
	"Target" and 75 terms	0.62 (142/228)

5. Conclusion

In this thesis, we propose a method to extract terms to be used with a topic term as search keywords for collecting tweets. First, we constructed a data set consisting of tweets relevant to a selected topic. Next, we extracted noun terms and then removed stopwords from the terms. Based on TF-IDF and PMI, to select more effective terms for search keywords, we considered TF-IDF*PMI score.

We selected four topics in which a lot of extraneous tweets occur when collecting tweets using a topic term such as “coach” as a search keyword and we extracted the top 15% terms using 150 relevant tweets for each topic, and then, we collected tweets for the topics using the terms with the topic term as multiple search keywords. As the result, we could collect tweets more accurately when using the topic term as a single search keyword.

Thus, we can say that the limited resources for crawling tweets such as search queries were used efficiently. Also, we could collect tweets relevant to the topic minimizing a waste of data storage space. Also, we can assume that the collected tweets are more helpful to extract meaningful information because they mostly consist of tweets relevant to the topics.

References

[1] Statista, http://www.statista.com/

[2] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo, “Earthquake shakes Twitter users: real-time event detection by social sensors”, In Proceedings of the 19th international conference on World wide web. ACM, 2010, pp. 851-860.

[3] Adam Bermingham and Alan F. Smeaton, “On Using Twitter to Monitor Political Sentiment and Predict Election Results”, Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP), IJCNLP, 2011, pp. 2-10.

[4] Meeyoung Cha, Hamed Haddadi, Fabr´ıcio Benevenuto, and Krishna P. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy", Proceedings of international AAAI Conference on Weblogs and Social, ICWSM, 2010, pp.10-17.

[5] Youngsub Han, Hyeoncheol Lee, and Yanggon Kim. “A Real-time Knowledge Extracting System from Social Big Data using Distributed Architecture”, Proceedings of the 2015 Conference on research in adaptive and convergent systems, ACM, 2015, pp. 74-79.

[6] Chengzhi Zhang, Huilin Wang, Yao Liu, Dan Wu, Yi Liao, and Bo Wang, “Automatic Keyword Extraction from Documents Using Conditional Random Fields”, Journal of Computational Information Systems, 2008, pp.1169-1180.

[7] Brian Lott, “Survey of Keyword Extraction Techniques”, UNM Education, 2012.

[8] Sungjick Lee and Han-joon Kim “News Keyword Extraction for Topic Tracking”, Networked Computing and Advanced Information Management, 2008. NCM'08. Fourth International Conference on, IEEE, 2008, pp. 554-559.

[9] Manning, C. D., Raghavan, P., Schutze, H. (2008). "Scoring, term weighting, and the vector space model." Introduction to Information Retrieval. pp. 100.

[10] Gerlof Boumaf, "Normalized (Pointwise) Mutual Information in Collocation Extraction", Proceedings of the Biennial GSCL Conference, 2009.

[11] Egidio Terra and C. L. A. Clarke, "Frequency Estimates for Statistical Word Similarity Measures" Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics. 2003. pp. 165-172

[12] J. Tao, F. Zheng, A. Li and Y. Li. "Advances in Chinese Natural Language Processing and Language Resources", In Proceedings of the Speech Database and Assessments, 2009 Oriental COCOSDA International Conference, 2009, pp.13-18.

[13] C. Surabhi. "Natural Language Processing Future", In Proceedings of International Conference on Optical Imaging Sensor and Security, 2013, pp.1 – 3

[14] The Stanford Natural Language Processing Group, http://nlp.stanford.edu

[15] NAGWANI, Naresh Kumar; SINGH, Pradeep. "Weight similarity measurement model based, object oriented approach for bug databases mining to detect similar and duplicate bugs" In Proceedings of the International Conference on Advances in Computing, Communication and Control. ACM, 2009. pp. 202-207.

[16] Twitter Developers, https://dev.twitter.com/

Curriculum Vita

NAME: Jeongwoo Kim

[redacted]

[redacted]

DEGREE AND DATE TO BE CONFERRED: Master of Science, 2015

Secondary Education	
2012	National Institute for Lifelong Education (NILE)