JISTaP http://www.jistap.org
Journal of Information Science Theory and Practice

Research Paper
J. of infosci. theory and practice 2(3): 29-39, 2014
http://dx.doi.org/10.1633/JISTaP.2014.2.3.3

Study of Machine-Learning Classifier and Feature Set Selection for Intent Classification of Korean Tweets about Food Safety

Ha-Neul Yeom
Korea University of Science and Technology (UST)
Korea Institute of Science and Technology Information (KISTI), Republic of Korea
lucetesky@kisti.re.kr, lucetesky@gmail.com

Myunggwon Hwang
Korea Institute of Science and Technology Information (KISTI), Republic of Korea
mgh@kisti.re.kr, mg.hwang@gmail.com

Mi-Nyeong Hwang *
Korea Institute of Science and Technology Information (KISTI), Republic of Korea
mnhwang@kisti.re.kr

Hanmin Jung
Korea University of Science and Technology (UST)
Korea Institute of Science and Technology Information (KISTI), Republic of Korea
jhm@kisti.re.kr

ABSTRACT
In recent years, several studies have proposed making use of the Twitter micro-blogging service to track various trends in online media and discussion. In this study, we specifically examine the use of Twitter to track discussions of food safety in the Korean language. Given the irregularity of keyword use in most tweets, we focus on optimistic machine-learning and feature set selection to classify collected tweets. We build the classifier model using Naive Bayes & Naive Bayes Multinomial, Support Vector Machine, and Decision Tree Algorithms, all of which show good performance. To select an optimum feature set, we construct a basic feature set as a standard for performance comparison, so that further test feature sets can be evaluated. Experiments show that precision and F-measure performance are best when using a Naive Bayes Multinomial classifier model with a test feature set defined by extracting Substantive, Predicate, Modifier, and Interjection parts of speech.

Keywords: Twitter, Tweets, Machine-learning Feature, Text Classification

Open Access

Accepted date: September 7, 2014
Received date: June 8, 2014

***Corresponding Author:** Mi-Nyeong Hwang
Senior Researcher
Korea Institute of Science and Technology Information (KISTI)
Republic of Korea
mnhwang@kisti.re.kr

1. INTRODUCTION

Within the Twitter micro-blogging service, a 'Tweet' is considered to be the basic unit of composed text and is limited to 140 words, including blanks and symbols, regardless of language. Twitter users can connect to one another as 'followers' and exchange tweets freely. The recent emergence of smartphone clients has encouraged a vast array of users to express their opinions via tweets.[1]

In recent years, several studies have suggested analysis of Twitter feeds to track various realtime trends, including journalistic influence, political attitudes, the vectors of certain illnesses, and the analysis of symptoms and treatments in public health crises (Choi et al., 2014; Tumasjan et al., 2010; Lampos et al., 2010; Paul & Dredze, 2011).

Similarly, in this study, we have concentrated on using Twitter to track trends involving food safety, such as those reflecting outbreaks of food poisoning, satisfaction with school food services, etc. For this work, we collected tweets using keywords for food safety topics and analyzed their contents. Unfortunately, the keywords used to collect tweets do not typically carry over to the collected content, and a large number of irrelevant tweets must be filtered out.

To this end, we propose use of a machine-learning classifier model and feature set for classifying collected tweets. Among machine-learning algorithms, the Naive Bayes & Naive Bayes Multinomial (McCallum & Nigam, 1998; Androutsopoulos et al., 2000; Youn & McLeod, 2007), Support Vector Machine (Drucker et al., 1999; Youn & McLeod, 2007) and Decision Tree (Youn & McLeod, 2007) algorithms have shown the best performance for classifying texts. We compare these algorithms to find the best classifier model. To select an optimum feature set, we construct a basic feature set as a standard for comparison, and then test other feature sets against this basic set.

We perform preprocessing to remove slang words and symbols from collected tweets and to analyze text morphology prior to building the machine-learning classifier models and the basic and test feature sets. We then measure the performance of the classifier models by applying the basic and test feature sets. Figure 1 shows the overall experimental process.

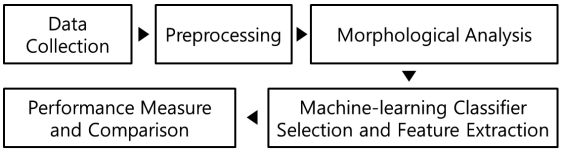

Fig. 1 Overall intent classification process

The rest of the paper is organized as follows: Section 2 presents related works; Section 3 describes our experimental process in detail; Section 4 provides an analysis of experimental results; and Section 5 contains concluding remarks.

2. RELATED WORKS

2.1. Machine Learning Algorithms for Text Classification

Text classification or document classification is just such a domain with a large number of attributes. The attributes of the examples to be classified are words, and the number of different words can be quite large indeed (McCallum & Nigam, 1998). In this section, we describe representative algorithms briefly.

2.1.1. Support Vector Machine (SVM)

Support Vector Machines (SVM) are a relatively new learning method used for binary classification. The basic strategy behind SVM is to find a hyperplane which separates the d-dimensional data perfectly into two classes (Dustin, 2002).

SVM generally exhibits slow training times but high accuracy, owing to its capacity to model complex nonlinear decision boundaries (margin maximization). It has been used both for classification and prediction and has been applied to handwritten digit recognition, visual object recognition, speaker identification, and benchmarking time-series prediction tests (Chen, 2010).

[1] Twitter (https://about.twitter.com/what-is-twitter/).

2.1.2. Naive Bayes and Naive Bayes Multinomial

All attributes of the examples are independent of each other given the context of the class. This is the so-called "Naive Bayes assumption." The Naive Bayes classifier is the simplest probabilistic classifier based on applying Naive Bayes assumption. While this assumption is clearly false in most real-world tasks, Naive Bayes often performs classification very well. Naive Bayes has been successfully applied to text classification in many research efforts.

The assumptions on distributions of features are called the event model of the Naive Bayes classifier. The Naive Bayes Multinomial classifier specifies that a document is represented by the set of word occurrences from the document. This approach has also been used for text classification by numerous people (McCallum & Nigam, 1998).

2.1.3. Decision Tree

A decision tree is a classifier expressed as a recursive partition of the instance space. Each internal node splits the instance space into two or more sub-spaces according to a certain discrete function of the input attributes values.

There are several advantages of the decision tree as a classification tool. First, decision trees are self–explanatory and when compacted they are also easy to follow. Second, decision trees can handle both nominal and numeric input attributes. Third, decision tree representation is rich enough to represent any discrete value classifier. Fourth, decision trees are capable of handling datasets that may have errors (Rokach & Maimon, 2005).

2.2. Korean Morphological Analyzer

The Korean morphological analyzer is a piece of software that analyzes the morphology of entered Korean text and outputs results as tagged parts of speech (POS). This tagged data plays a major role in Korean natural language processing (KAIST Semantic Web Research Center, 2011).

Hannanum is the most significant Korean morphological analyzer (Lee et al., 1999). In this paper, we use the Hannanum morphological analyzer to process the morphology of tweets and to tag each word by its POS.

The following two sections describe relevant portions of Korean grammar and provide details on the Hannanum morphological analyzer.

2.2.1. Korean Grammar[2]

Nine POS are distinguished in Korean: noun, pronoun, numeric, verb, adjective, determiner, adverb, interjection, and postposition. These are further grouped into five categories: Substantive, predicate, modifier, interjection, and postposition.

(1) Substantive: Includes nouns, pronouns, and numerics, and grouped into independent noun and dependent noun classes. A dependent noun can be used with an adnominal phrase. The dependent noun can be used independently in a sentence with the help of a postposition, or to predicate and modify other POS. A substantive takes a case marker and can modify a determiner. In this paper, we call this category 'Substantive' to distinguish it from nouns proper.
(2) Predicate: Includes verbs and adjectives. A predicate is used to predicate the action or property of a subject, and is inflected by a suffix. In this paper, we call this category 'Predicate.'
(3) Modifier: Includes determiners and adverbs. A modifier is only used to modify a substantive or predicate, and does not take a case marker or suffix.
(4) Interjection: Used independently in the presence of a sentence. It is not predicated, and does not predicate other POS. Of the nine POS mentioned above, only interjections belong to this category, so we call it simply 'Interjection.'
(5) Postposition: Used to present grammatical relations between words. Of the nine POS mentioned above, only postpositions belong to this category, so we call it simply 'Postposition.'

2.2.2. Hannanum Morphological Analyzer

The Hannanum Morphological Analyzer is de-

[2] Doosan Encyclopedia (http://terms.naver.com/entry.nhn?docId=1189685&cid=40942&categoryId=32978).

veloped by KAIST Semantic Web Research Center.[3] Hannanum users can control the morphological analysis process by selecting plug-ins and constructing a workflow. Workflow is composed of three phases: The preprocessing phase, the morphological analysis phase, and the POS tagging phase. The plug-ins performing each of these phases are classified into two groups: Major plug-ins and supplemental plug-ins. Major plug-ins perform primary functions like morphological analysis and POS tagging, whereas supplemental plug-ins perform functions like recognizing sentences, converting morphology tags, extracting nouns, etc. (KAIST Semantic Web Research Center, 2011).

3. METHOD FOR SELECTING FEATURE SETS BY MACHINE-LEARNING ALGORITHM

3.1. Data Collection and Preprocessing

We begin by collecting tweets based on keywords targeting food safety from twitter feeds covering the period from January 2014 until some recent date. As of April 30 of this year, the raw tweets collected by this process numbered around 15 million. From this collected set, we narrow our test dataset to tweets that include 'Geupsig' (food service). This test dataset includes tweets with a wide range of attitudes toward free school meals, such as those including abusive language (e.g., 'Geupsigsaeggi').

We classify the test data into a positive set and a negative set. The subjects of tweets classified in the positive set concern the evaluation of food services. Once this set is classified, the remainder of the tweets is classified as the negative set. Table 1 below shows the size of the positive, negative, and total sets used in our experiment.

Table 1. Size of Positive, Negative and Total Sets used in Experiment

Positive	Negative	Total
500	500	1000

Among the collected tweets are a large number of 'retweets' (i.e. duplicate tweets) and news links unrelated to our analysis. Once these are removed, slang words that use symbols to reflect users' attitudes are removed too. Note that these words generally mark a tweet for classification in the negative set.

3.2. Morphological Analysis (Lee et al., 1999)

We use the Hidden Markov Model (HMM) POS Tagger Workflow included in the Hannanum morphological analyzer to analyze the tweets in the test dataset. The HMM POS Tagger Workflow is composed of the SentenceSegmentor, InformalSentenceFilter, ChartMorphAnalyzer, UnknownProcessor, and HMMTagger.

A description of the analysis process and the characteristics of each plug-in are as follows.

① SentenceSegmentor: Segments sentences based on delimiters like periods, question marks, and exclamation marks.

② InformalSentenceFilter: Filters noise based on patterns. The patterns are learned from a dataset of over 90,000 replies.

③ ChartMorphAnalyzer: Uses lattice storage to analyze morphology.

④ UnknownProcessor: Handles words that are not found in the dictionary.

⑤ HMMTagger: Tags POS based on a Hidden Markov Model. Tagging reflects dependencies between separate words and morphology.

3.3. Machine-learning Classifier Selection and Feature Extraction

Our goal is to find an optimistic machine-learning classifier model and a feature set for classifying the intents of collected tweets. For this, we use the Naive Bayes & Naive Bayes Multinomial, Support Vector Machine, and Decision Tree methods, and compare their performance.

To find an optimistic feature set, we construct a basic feature set as a standard for comparison and then a number of test feature sets. The basic feature set is the result of simple preprocessing. Test feature sets are created by extracting words corresponding to a particular POS from the basic feature set.

[3] KAIST Semantic Web Research Center (http://semanticweb.kaist.ac.kr/home/index.php/HanNanum).

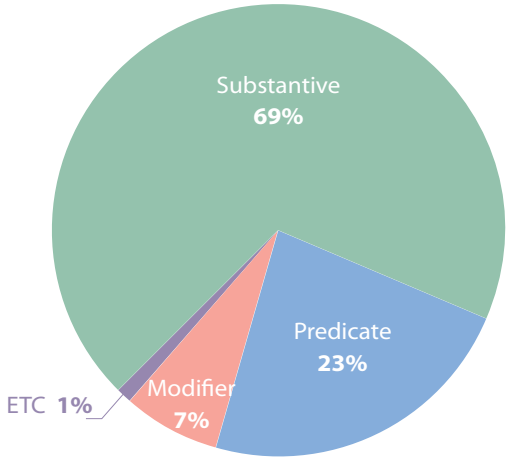

Fig. 2 Component ratios for extracted POS in test sets

We calculate the component ratio of POS in each test set. Figure 2 shows the component ratios for the extracted POS in our test sets (Substantive: 69%; Predicate: 23%; Modifier: 7%; Other POS: 1%). Thus we create a test feature set by extracting Substantive, Predicate, and Substantive-Predicate that have higher percentages than the others (feature B, D, F). Feature C, E, G are created by adding modifier and interjection to feature B, D, F because modifiers and interjections account for a small percentage. Table 2 describes the basic feature set (A), test feature set (B~H), and examples of each feature set.

We propose use of a machine-learning classifier model and feature set for classifying collected tweets. So we track precision, recall, and F-measure. Precision measures how accurate a machine-learning classifier model is, and recall measures how many tweets can be classified by a machine-learning classifier model. F-measure measures reliability of precision and recall. Lastly, to increase reliability of measuring classifier model performance, we measure the performance of the classifier models using Weka[4] with ten-fold cross validation.

Table 2. Basic Feature Set (A), Test Feature Set (B~H), and Examples of Each Feature Set

	Feature Set	Example of Feature Extraction
A	Preprocessing	gieogeul, doesaegyeobomyeon, geupsigiraneun, mulgeoneun, geunyang, baereul, chaeugi, wihae, meogneungeoji, masisseurago, mandeun, mulgeoneun, anida, eoddeoghaeya, giseongmandureul, maseobsge, mandeulsuga, isseulgga
B	Substantive	gieog, geupsig, mulgeon, bae, meogneungeoji, mulgeon, giseong, mandu, su
C	Substantive, Modifier, Interjection	gieog, geupsig, mulgeon, geunyang, bae, meogneungeoji, mulgeon, giseong, su
D	Predicate	doesaegi, chaeu, wiha, masiss, mandeul, ani, eoddeogha, maseob, mandeul, iss
E	Predicate, Modifier, Interjection	doesaegi, geunyang, chaeugi, wiha, masiss, mandeul, ani, eoddeogha, maseob, mandeul, iss
F	Substantive, Predicate	gieok, doesaegi, geupsik, mulgeon, bae, chaeu, wiha, meokneungeoji, masiss, mandeul, mulgeon, ani, eoddeogha, giseong, mateob, mandeul, iss
G	Substantive, Predicate, Modifier, Interjection	gieok, doesaegi, geupsik, mulgeon, geunyang, bae, chaeu, wiha, meokneungeoji, masiss, mandeul, mulgeon, ani, eoddeogha, giseong, maseob, mandeul, iss
H	Modifier, Interjection	geunyang

[4] Weka is a collection of machine-learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from the user's own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well suited for developing new machine-learning schemes (Hall et al., 2009).

4. EXPERIMENTAL RESULTS

The figures and tables below show the precision, recall, and f-measure of each classifier model and each feature set.

When applying the Naive Bayes classifier model to feature sets for Substantive, Substantive-Modifier-Interjection, Substantive-Predicate, and Substantive-Predicate-Modifier-Interjection categories (B, C, F, G), F-measure is 6.4~6.8% higher than for the basic feature set. Table 3 and Figure 3 show the performance of the Naive Bayes classifier, respectively.

Table 3. Performance of Naive Bayes Classifier Model

Feature Set	Precision	Recall	F-Measure
A	0.822	0.774	0.797
B	0.837	**0.894**	**0.865**
C	0.839	0.886	0.862
D	0.748	0.718	0.733
E	0.770	0.684	0.725
F	0.851	0.870	0.861
G	**0.861**	0.868	**0.865**
H	0.591	0.710	0.645

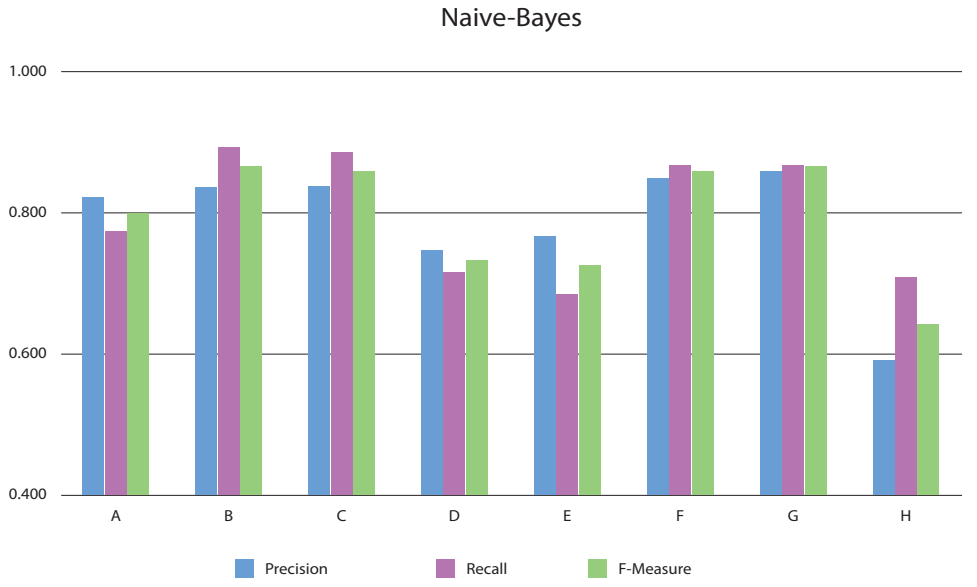

Fig. 3 Graph of performance of Naive-Bayes classifier model

Similarly, when applying the Naive Bayes Multinomial classifier model to feature sets for Substantive-Modifier-Interjection, Substantive-Predicate, and Substantive-Predicate-Modifier-Interjection categories (C, F, G), the performance is higher than that of the basic feature set. When applied to the test feature set for Substantive, precision is 5.7% lower than for the basic feature set, but F-measure and recall for feature set B are 16.3% and 35.4% higher than for the basic feature set. Precision for the test feature sets, including Substantive-Predicate and Substantive-Predicate-Modifier-Interjection (F, G) is about 90%. This is the best overall precision result for the Naive Bayes Multinomial classifier model. Table 4 and Figure 4 show the performance of the Naive Bayes Multinomial classifier, respectively.

Table 4. Performance of Naive-Bayes Multinomial Classifier Model

Feature Set	Precision	Recall	F-Measure
A	0.874	0.622	0.727
B	0.817	**0.976**	0.890
C	0.887	0.924	0.905
D	0.716	0.634	0.672
E	0.718	0.668	0.692
F	0.901	0.928	0.914
G	**0.906**	0.930	**0.918**
H	0.479	0.322	0.385

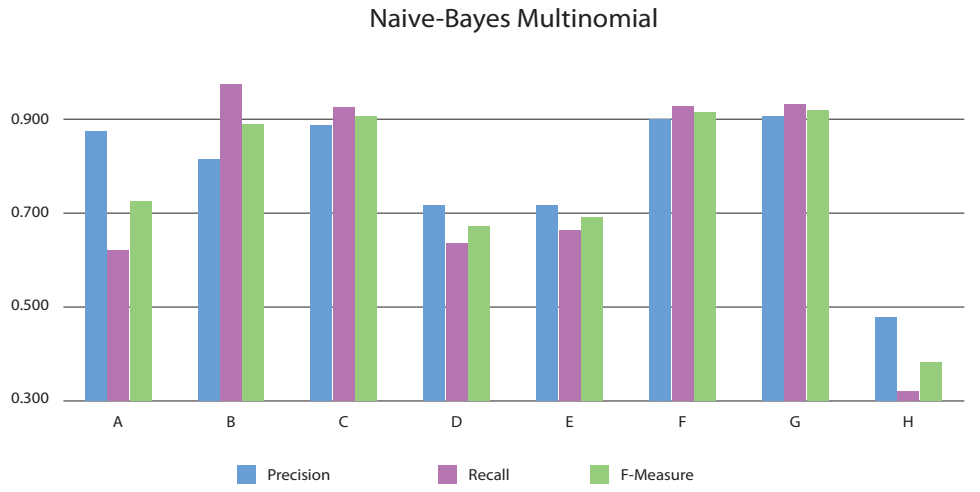

Fig. 4 Graph of performance of Naive-Bayes Multinomial classifier model

When using the Support Vector Machine and Decision Tree classifier models, precision for the basic feature set is higher than for all other feature sets, whereas recall rates for all test feature sets were higher than the recall for the basic feature set. In particular, F-measure and recall rates for test feature sets for Substantive, Substantive-Modifier-Interjection, Substantive-Predicate, and Substantive-Predicate-Modifier-Interjection categories are up to 11.3% and 27% higher than for the basic feature set. Table 5 and 6 and Figure 5 and 6, respectively, show the performance of the Support Vector Machine and the Decision Tree classifier.

Table 5. Performance of Support Vector Machine Classifier Model

Feature Set	Precision	Recall	F-Measure
A	**0.889**	0.740	0.808
B	0.819	**0.962**	0.885
C	0.874	0.860	0.867
D	0.698	0.840	0.762
E	0.698	0.814	0.752
F	0.883	0.892	0.888
G	0.884	0.900	**0.892**
H	0.584	0.790	0.672

Table 6. Performance of Decision Tree Classifier Model

Feature Set	Precision	Recall	F-Measure
A	**0.860**	0.666	0.751
B	0.772	0.932	0.844
C	0.785	0.920	0.847
D	0.665	0.790	0.722
E	0.670	0.780	0.721
F	0.803	**0.936**	**0.864**
G	0.801	0.920	0.857
H	0.528	0.862	0.655

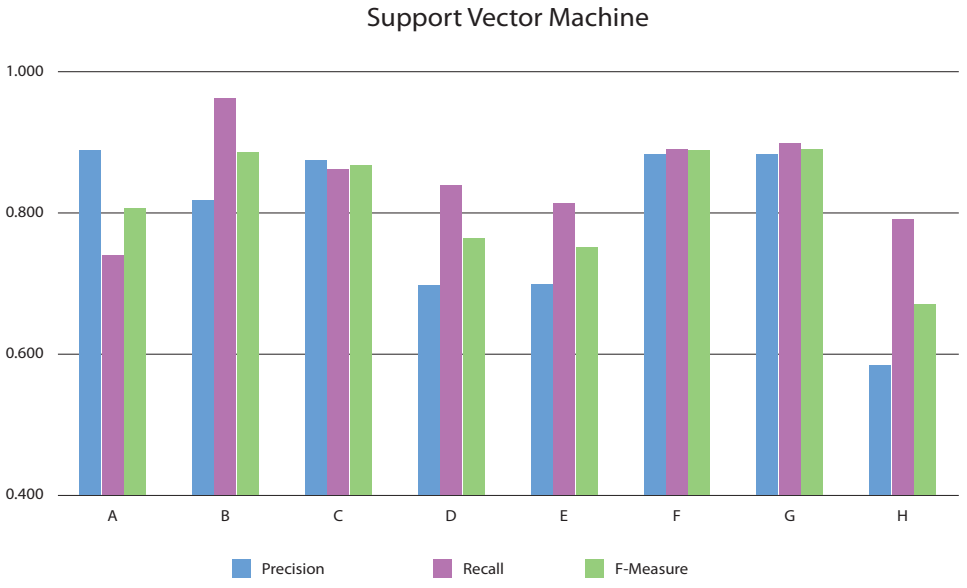

Fig. 5 Graph of performance of Support Vector Machine

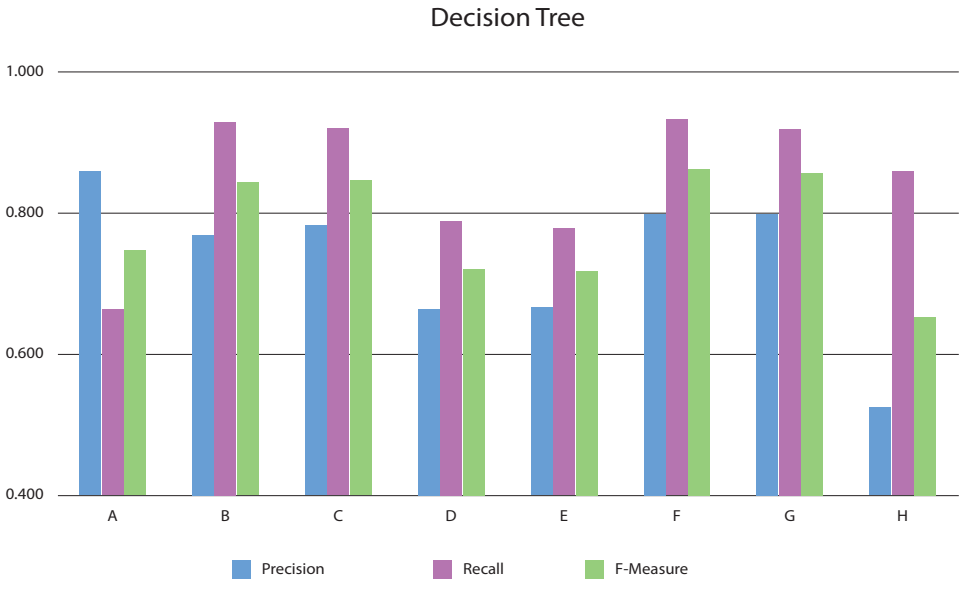

Fig. 6 Graph of performance of Decision Tree classifier model

Overall, we conclude that the Naive Bayes Multinomial classifier model outperforms other models.

Normally, it is assumed that the mor e words in a feature set, the better the precision will be. Our experiments yield a different result: The F-measure and precision rates for test feature sets for Substantive or Substantive-Predicate categories (B, C, F, G) are higher than for the basic feature set. Furthermore, recall rates for test feature sets for Substantive-Modifier-Interjection, Substantive-Predicate, and Substantive-Predicate-Modifier-Interjection categories are up to 35% higher than the recall for the basic feature set.

5. CONCLUSIONS

We proposed the use of a machine-learning classifier model and feature set for classifying collected tweets.

In experiments, F-measure and precision rates for test feature sets of Substantive or Substantive-Predicate categories (B, C, F, G) are higher than that for the basic feature set. This result shows that using many words does not guarantee more accurate classification.

Recall rates for the test feature sets are far higher than that for the basic feature set. Particularly, when using the Naive Bayes Multinomial classifier model, recall rates for test feature sets including Substantive-modifier-interjection, Substantive-predicate, and Substantive-predicate-modifier-interjection categories are about 30% higher than that for the basic feature set. The recall for test feature set B using only Substantive is the best in each classifier model except for Decision Tree. The reason is that Substantive and Predicate categories deal with different words depends on suffixes (endings of words). So recall for the test feature set is far higher than that for the basic feature set.

When using the Naive Bayes Multinomial classifier model and applying test feature set G (Substantive, Predicate, Modifier, Interjection), the precision is the best (90.6%). Furthermore, when using the same classifier, the recall for test feature set B (Substantive) is the best (97.6%). But the precision for the same test feature set is 6% lower than recall for the basic feature set (81.7%). F-measure for Naive Bayes Multinomial classifier model and test feature set G (Substantive, Predicate, Modifier, Interjection) is the best (91.8%, Precision 90.6%, Recall 93%).

In our final analysis, the Naive Bayes Multinomial classifier and test feature set G (Substantive-Predicate-Modifier-Interjection) proved to be the best pairing for classification of Korean tweets about food safety.

In the future, to increase reliability of measuring classifier model performance, we will use more test data sets and analyze morphology more accurately. And we will use classifier models to classify intent values of the rest of the collected tweets.

ACKNOWLEDGEMENTS

This work was supported by the IT R&D program of MSIP/KEIT [2014-044-024-002, Developing On-Line Open Platforms to Provide Local-Business Strategy Analysis and User-Targeting Visual Advertisement Materials for Micro-Enterprise Managers].

REFERENCES

Androutsopoulos, I., Koutsias, J., Chandrinos, K. V., Paliouras, G., & Spyropoulos, C. D. (2000). An evaluation of Naive Bayesian anti-spam filtering. Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning, 9-17.

Chen, B. (2010). Chapter 6. Classification and prediction. Lecture Note Distributed in Data Mining CSCI 4370/5370 at Georgia State University, Retrieved June 2, 2014, from http://storm.cis.fordham.edu/~yli/documents/CISC4631Spring12/Chapter6_Class1.ppt.

Choi, D., Hwang, M., Kim, J., Ko, B., & Kim, P. (2014). Tracing trending topics by analyzing the sentiment status of Tweets. *Computer Science and Information Systems, 11*(1), 157-169.

Drucker, H., Wu, D., & Vapnik, V. N. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural Networks, 10*(5), 1048-1054.

Dustin, B. (2002). Introduction to support vector machines. Retrieved Jun 2, 2014, from https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0CCYQFjAA&url=http%3A%2F%2Fwww.work.caltech.edu%2F~boswell%2FIntroToSVM.pdf&ei=4f-

6SU9P2A4K48gW6noHIBw&usg=AFQjCNGlfz-DO-ZpOtj219pI81FgjP2yyEA&sig2=SdwK6M-V4e2EVzFaZuZhLEw

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: An update. ACM SIGKDD *Explorations Newsletter, 11*(1), 10-18.

KAIST Semantic Web Research Center, (2011). Hannanum Korean Morphological Analyzer User Manual.

Lampos, V., Bie, T. D., & Cristianini, N. (2010). Flu detector-tracking epidemics on Twitter. *Machine Learning and Knowledge Discovery in Databases, 6323*, 599-602.

Lee, W., Kim, S., Kim, G., & Choi, K. (1999). Implementation of modularized morphological analyzer. Proceedings of Korean Institute of Information Scientists and Engineers: Special Interest Group on Human Language Technology, 123-136.

McCallum, A., & Nigam, K. (1998). A comparison of event models for Naive Bayes text classification. AAAI-98 Workshop on Learning for Text Categorization, 752, 41-48.

Paul, M. J., & Dredze, M. (2011). You are what you Tweet: Analyzing Twitter for public health. Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media.

Rokach, L., & Maimon, O. (2005). Decision trees. *Data mining and knowledge discovery handbook*. Springer, US, 165–192.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, 178-185.

Youn, S., & McLeod, D. (2007). A comparative study for email classification. Advances and Innovations in Systems, Computing Sciences and Software Engineering, Springer, Netherland, 387-391.