**Original Article**

Healthc Inform Res. 2011 December;17(4):224-231.
http://dx.doi.org/10.4258/hir.2011.17.4.224
pISSN 2093-3681 • eISSN 2093-369X

HIR
Healthcare Informatics Research

# A Multi-Classifier Based Guideline Sentence Classification System

**Mi Hwa Song, PhD[1], Sung Hyun Kim, MA[2], Dong Kyun Park, MD, PhD[3], Young Ho Lee, PhD[1]**

[1]U-Healthcare Institute, Gachon University of Medicine and Science, Incheon; [2]LNISOFT, Incheon; [3]U-Healthcare Center, Gachon University Gil Hospital, Incheon, Korea

**Objectives:** An efficient clinical process guideline (CPG) modeling service was designed that uses an enhanced intelligent search protocol. The need for a search system arises from the requirement for CPG models to be able to adapt to dynamic patient contexts, allowing them to be updated based on new evidence that arises from medical guidelines and papers. **Methods:** A sentence category classifier combined with the AdaBoost.M1 algorithm was used to evaluate the contribution of the CPG to the quality of the search mechanism. Three annotators each tagged 340 sentences hand-chosen from the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure (JNC7) clinical guideline. The three annotators then carried out cross-validations of the tagged corpus. A transformation function is also used that extracts a predefined set of structural feature vectors determined by analyzing the sentential instance in terms of the underlying syntactic structures and phrase-level co-occurrences that lie beneath the surface of the lexical generation event. **Results:** The additional sub-filtering using a combination of multi-classifiers was found to be more effective than a single conventional Term Frequency-Inverse Document Frequency (TF-IDF)-based search system in pinpointing the page containing or adjacent to the guideline information. **Conclusions:** We found that transformation has the advantage of exploiting the structural and underlying features which go unseen by the bag-of-words (BOW) model. We also realized that integrating a sentential classifier with a TF-IDF-based search engine enhances the search process by maximizing the probability of the automatically presented relevant information required in the context generated by the guideline authoring environment.

**Keywords:** Knowledge Bases, Data Mining, Natural Language Processing

## I. Introduction

Clinical process guidelines (CPG) are an effective tool for minimizing the gap between a physician's clinical decision and medical evidence and for modeling the systematic and standardized pathway used to provide better medical treatment to patients [1]. The CPG modeling service encodes clinical knowledge for solving problems and is used to create a flow diagram that models the whole process of the clinical event structure, thereby allowing the inference engine to use the knowledge base and clinical algorithm [2]. It should also provide a way of updating existing rules and algorithms to better reflect the dynamic context of patients and refine them

**Submitted:** September 23, 2011
**Revised:** October 21, 2011
**Accepted:** December 5, 2011

**Corresponding Author**
Young Ho Lee, PhD
U-Healthcare Institute, Gachon University of Medicine and Science, 534-2 Yeonsu-dong, Yeonsu-gu, Incheon 406-799, Korea. Tel: +82-32-820-4100, Fax: +82-32-820-4109, E-mail: leeyh@gachon.ac.kr

based on new medical, scientific findings. While encoding rules, creating algorithms, and updating the knowledge base, there naturally arises the need for searching and administering relevant knowledge. Perceiving and approximating the needs structure, we designed an integrated architecture that aggregates knowledge, searching, authoring, and administration of knowledge within a single presentation layer. Further understanding physicians' internal needs, we decided to embed a sentential classifier that automatically presents information that the physician may want to find, in a better format, based on the context of the CPG-authoring events in the user interface. To enhance sentence classification accuracy, we employed dimensionality reduction by feature extraction and ensemble learning in which weak classifiers are sequentially combined to form a committee of experts by the AdaBoost.M1 meta-algorithm. Previous studies on the sentence classification, multi-classifier-based categorization, and feature representation are summarized.

## 1. Sentence Classification

There has been considerable related research on sentential classification, especially in spam mail filtering and biomedical text mining. Pan [3] has done extensive research in biomedical sentence classification in which multi-label tagging is done, rather than single-label-based tagging. The basic presupposition in the study is that a sentence is an instance generated by a set of hypothetical classes such as Focus/Polarity/Certainty/Evidence/Direction or Trend. Although the study paved the way for an advanced sentential classification problem, it does not report the effect of employing ensemble learning. Xin et al. [4] reports the effect of using classifier combining in a sentence classifier used for a Q&A system by implementing the AdaBoost.M1 [5,6] algorithm, a version of Adaptive Boosting. Using combined multiple classifiers in a Q&A system is not reported to be a conspicuously positive factor in increasing classification accuracy. A single classifier suffices because the problem domain is not rich in long sentences and spoken sentences prevail.

Contrary to the domain mentioned above, clinical guideline texts, tend to be long and to face the problem of classifying an instance generated within the high-dimensional feature space, also known as "the curse of dimensionality" [7], especially when adopting a Term Frequency-Inverse Document Frequency (TF-IDF) [8]-based vector space model. The problem arises because each Boolean feature tends to be scattered extensively in the feature space. In other words, the Boolean feature is the existence of a token that characterizes the instance to which it belongs. The classification of a sentence category is especially problematic, because a sentence is relatively short compared with a document. This causes even semantically similar instances to occupy highly scattered cluster areas in the feature space.

Given this view of the problem, the main points of our research are the use of: 1) the transformation function that extracts the predefined set of structural feature vectors by analyzing the sentential instance in terms of the underlying syntactic structure and phrase-level co-occurrence that lie beneath the surface of the lexical generation event; and 2) ensemble learning, which is being increasingly adopted in diverse pattern recognition applications, to tackle the difficulty inherent in classifying instances generated in noisy, high-dimensional and non-linear systems, like textual objects.

## 2. Multi-Classifier-based Categorization

The concept behind using a multi-classifier is to train "a committee of experts," each of whom specializes in a subportion of data points in the feature space, since a single classification model can't classify all of the data points without generating classification errors, either in training or in test. The sequential combination of a weak classifier continues until there is none misclassified, or the rate of classification success converges to some satisfactory level. When the training of multiple classifiers using this meta-training algorithm completes, we get N experts for N sub-sets of the whole data set. The aim of the boosting meta-algorithm is to learn the optimal parameter of the robust non-linear hyperplane by sequentially combining a set of even heterogeneous classification models whose optimal hypothesis is determined to minimize the error. Since the meta-algorithm allows heterogeneous training algorithms to be combined, we experimented with various approaches, for example, Naïve Bayes [9], Support Vector Machine [10], Maximum Entropy [11,12], Multi-layered Perceptron, and Radial Basis Function Network [13,14].

## 3. Feature Representation

Transforming an instance or event into an array of values, either numeric or nominal, is called feature representation and the array of the values is called the feature vector, which is an input to the classifier. In general, the bag-of-words (BOW) [15] approach is taken to extract the feature vector in a textual object classification problem. Using the whole set of features in the BOW approach is intractable due to its high dimensionality. This is especially problematic when the textual object is a sentence which has limited features taken in its instantiation out of quite large feature set, which means that in extreme cases even two semantically

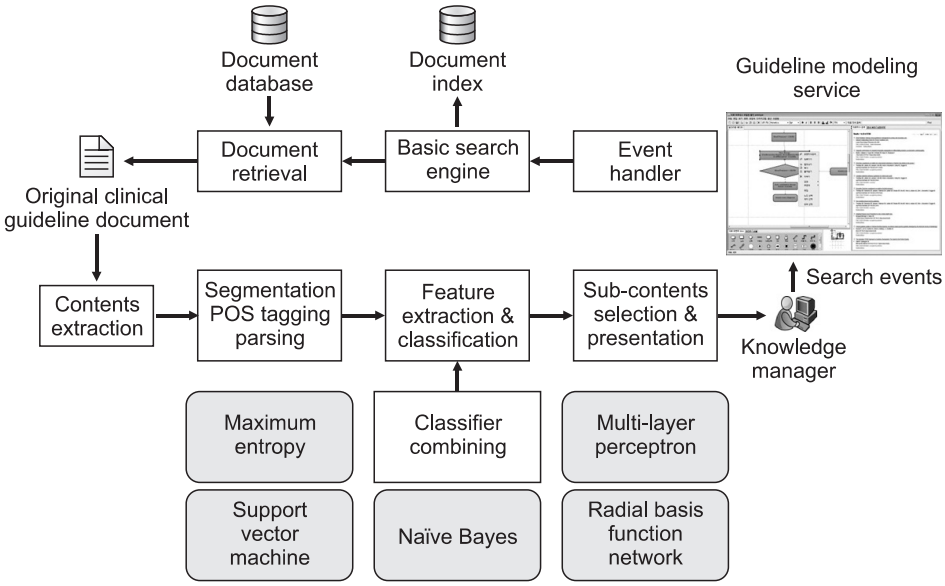


Figure 1. Searching clinical evidence enhanced with a sentential classifier. POS: part-of-speech.

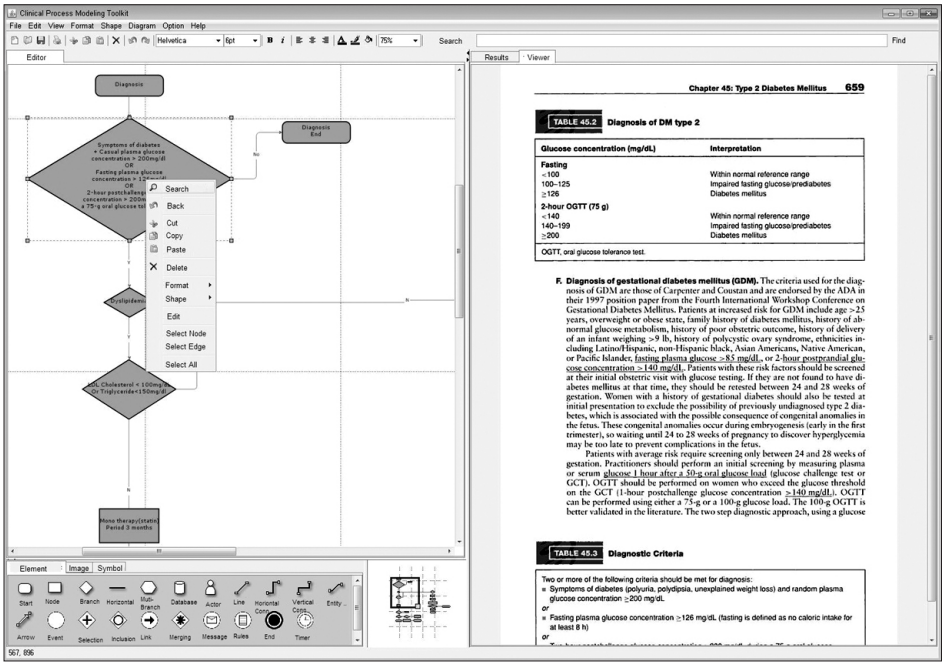


Figure 2. A snapshot of the system user interface.

similar instances could consist of mutually-exclusive Boolean feature sets. This is where the dimensionality reduction approach comes in. To reduce the size of the feature dimension, feature selection algorithms such as linear discriminant analysis (LDA) [16] or information gain (IG) [17] are used to filter out some irrelevant features that do not contribute much to the discrimination of instances. Another approach to cope with the problem of high dimensionality is to use a transformation function that captures some predefined set of mappings from the superficial features to some structural features of the generative event working in the instantiation of a sentence object. For example, a named entity, part-of-speech (POS) or the number of phrasal co-occurrences within a sentential construction is not directly considered in the BOW approach. To take such features into consideration, we designed a set of transformation functions, that is, feature extractors [18], that use structural analysis of a sentence to create an output set of real values. This can be understood as a mapping from a high dimensional qualitative space to a low dimensional quantitative one.

## II. Methods

### 1. Proposed System

Our sentence classifier has been designed to be embedded in the knowledge base authoring services for clinical deci-


http://dx.doi.org/10.4258/hir.2011.17.4.224

sion support systems that use rule-based inference engines. When a medical domain expert models the clinical pathway or clinical rules, he or she should need some evidence for encoding domain knowledge. Therefore, we decided to integrate a search function into the authoring service. To build the search system, the Nutch web crawler is used to aggregate and routinely crawl medical documents for patients with chronic disease [19]. Tika is used for extracting text and meta-information within a PDF file when the Lucene-based indexer creates the search index [20]. The natural language processing component is built upon the sentence splitter and segment recognizer from OpenNLP [21] and the Stanford Parser [22] is used in the preprocessing step for extracting the structural feature vector from a sentential instance.

The overall scenario is as follows. When a medical domain expert is creating a flow diagram using the authoring tool, he or she may need some relevant information concerning a rule represented in a node in the process graph and click on the node to pop up a contextual menu that contains the "Search Relevant Guideline." A handler listens to the click event and creates a search process and a bunch of search results gets listed, upon one of which the user clicks to get directly to the page that contains the clinical guideline relevant to the node. A sentence classifier categorizes each sentence in a PDF document to find the page that contains the largest number of a specific sentence class, in our case, the FRS tag, whose description is given in the next section (Figures 1, 2).

## 2. Training Data Preparation

We use a single-label-based classifier, contrary to the experiment done by Shatkay et al. [23], who adopts a multi-label-based classifier. Although multi-label-based classification is theoretically plausible, creating a training set with a multi-label tag was not a good choice for our application, which enhances the search process by providing users with the function to automatically locate the page with certain information. Additionally, using training data made by human annotators doesn't guarantee that the corpus tagged with a multi-label is noise-free, that is, without errors, since even a single sentential instance in practice is prone to be tagged differently by different annotators. As a consequence, there may be a very low rate of agreement among multiple annotators on their tagging of a sentence with multi-label tagging. The Report of the Joint National Committee on Prevention,

Table 1. Four sentence classes based on semantic function

| Category | Sentence examples |
|---|---|
| FRS | After BP is at goal and stable, follow-up visits can usually be scheduled at 3- to 6-month intervals. |
| RECOMMEND | Serum potassium and creatinine should be monitored at least one to two times per year. |
| ANALYSIS | Patients with occlusive CAD and/or LVH are put at risk of coronary events if DBP is low. |
| GENERAL | Prevention and lifestyle modifications for overweight and obesity. |

FRS: formal representation string, BP: blood pressure, CAD: coronary artery disease, LVH: left ventricular hypertrophy, DBP: diastolic blood pressure.

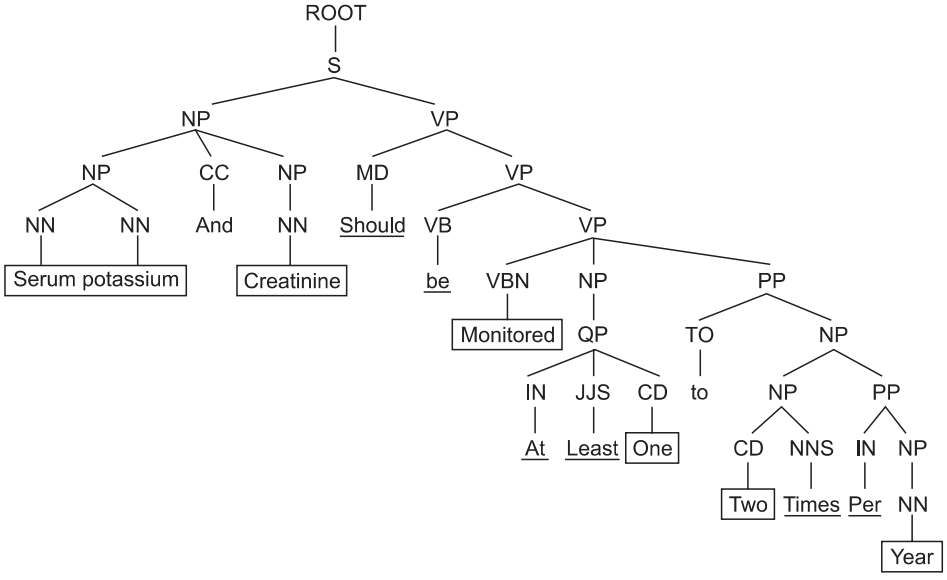


Figure 3. Parsed sentence from which to extract the feature vector. CC: coordinating conjunction, CD: cardinal number, IN: preposition or subordinating conjunction, JJS: adjective, superlative, MD: modal, NN: noun, singular or mass, NP: noun phrase, NNS: noun, plural, PP: prepositional phrase, QP: quantifier phrase, S: sentence, TO: to, VB: verb, base form, VBN: verb, past participle, VP: verb phrase.

Table 2. The description of each feature by transformation function

| Feature # | Description |
|---|---|
| 1 | The number of symbolic tokens which occurred in a formal representation sentence that encodes certain clinical situation or rules. |
| 2 | The number of phrasal expressions which occurred in a formal representation sentence. |
| 3 | The number of co-occurrence events of tokens which occurred in a formal representation sentence. (Named entity is recognized as a single token.) |
| 4 | The number of phrasal expressions which occurred in a <RECOMMEND> sentence. |
| 5 | The number of phrasal expressions which occurred in an <ANALYSIS> sentence. |

Detection, Evaluation, and Treatment of High Blood Pressure (JNC7) [24] is used for source of guideline text. Our training data is tagged in the manner shown below Table 1.

<FRS> stands for "Formal Representation String" and means the sentence that includes the clinical rule in a guideline document. Although formal representation is basically constructed on a set of controlled vocabulary and numerical symbols to make expression clear and disambiguated, some authors follow such a rule and others use their own style of expression allowed by a natural language. Therefore, we use <FRS> to classify whatever can semantically be classified as a clinical rule, regardless of whether or not a given sentence is actually a formal representation. The <RECOMMEND> tag is attached to the sentences that recommend some treatment for patients satisfying some set of clinical conditions. <ANALYSIS> is used for the sentences with some expressions conveying scientific or statistical facts induced from either experimental research on a cohort data set or observational test of patients. <GENERAL> is a tag for the sentences semantically included in any other collective classes than specified above. Three annotators each tagged 340 sentences taken from a clinical guideline text (JNC7) by hand. Then three annotators carried out cross-validation of the tagged corpus.

## 3. Training Data Representation and Feature Extraction

The feature extractor is designed to exploit the features of the underlying event structure at the point of sentential instantiation whose generative process includes pattern template and phrasal co-occurrence. The pattern template captures some structural characteristics from syntactic hierarchy and repetition of in-domain idioms. A phrasal co-occurrence event takes place for the sentence to express certain domain situations or facts and there may be either short or long range dependency within a sentence. Figure 3 illustrates some points to be explained. If we depended only on the BOW model of sentential categorization, it would not be possible to transform the (sentence generation) event

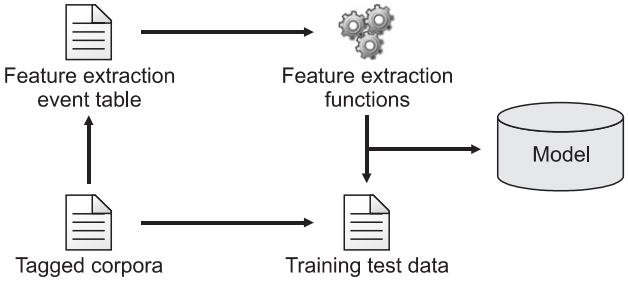


Figure 4. Creating feature extractors.

structure into a set of features to be considered in inducing the more generalized discriminant. The lexical feature, which is Boolean by nature, cannot explain the co-occurring and repetitive template-based features like [should + be + VBN]. Therefore, replacing the token "monitored" in the test set with any other past participle (VBN), like "observed" or a phrase, such as "periodically tested", may ignore the structural similarity determined by their being within the same semantic cluster in low dimensional feature space. Since the transformation function reveals generic structure across the in-domain event set, whereas the lexical feature is specific to individual data, using transformation functions that capture such structural features may decrease the probability of learning over-fitting model parameters. This means that certain lexical events may take place in the training set and may not exist in the test set at all.

The Stanford Parser was used to analyze the natural language event structure. Based on the parse tree output, the transformation function outputs five dimensional feature vectors, all of them being real valued. The description of each feature captured by the transformation follows in Table 2.

For example, the real valued feature vector extracted from the parse tree in Figure 3 is as follows: [0 0 3 3 2] → <RECOMMENDED>.

The first value represents the fact that the number of symbol tokens such as <, >, or mL, frequently used in formal representation in clinical rules, equals 0 in a sentence instance. The second value represents the fact that the phrasal

 http://dx.doi.org/10.4258/hir.2011.17.4.224

Table 3. Feature event table

| FRS | | RECOMMEND | |
|---|---|---|---|
| Phrasal event | Ratio | Phrasal event | Ratio |
| At least CD | 10/80 | At least CD | 70/80 |
| CD years of age | 10/15 | Should be VBN | 120/175 |
| BMI > CD kg/$m^2$ | 9/10 | Necessary to VB | 130/150 |
| Stroke or transient | 4/6 | Recommended that the patient VBZ | 2/5 |

FRS: formal representation string, CD: cardinal number, VBN: verb, past participle, BMI: body mass index, VB: verb, base form, VBZ: verb, 3rd person singular present.

Table 4. Performance result of each classifier

| Method | Precision | Recall | F-measure | ROC area |
|---|---|---|---|---|
| NB | 0.825 | 0.774 | 0.784 | 0.926 |
| MaxEnt | 0.815 | 0.797 | 0.800 | 0.926 |
| SVM | 0.812 | 0.794 | 0.797 | 0.863 |
| RBFN | 0.818 | 0.794 | 0.799 | 0.929 |
| MPerceptron | 0.811 | 0.791 | 0.794 | 0.925 |

ROC: receiver operating characteristic, NB: Naïve Bayes, MaxEnt: maximum entropy, SVM: support vector machine, RBFN: radial basis function network, MPerceptron: multi-layer perceptron.

Table 5. Combination of multi-layered perceptron classifiers by AdaBoost.M1

| Training order | Method | Precision | Recall | F-measure | ROC area |
|---|---|---|---|---|---|
| 1st | MPerceptron | 0.811 | 0.791 | 0.794 | 0.925 |
| 2nd | MPerceptron | 0.815 | 0.797 | 0.800 | 0.910 |
| 3rd | MPerceptron | 0.815 | 0.797 | 0.800 | 0.924 |
| 4th | MPerceptron | 0.815 | 0.797 | 0.800 | 0.924 |
| 5th | MPerceptron | 0.815 | 0.797 | 0.800 | 0.924 |
| 6th | MPerceptron | 0.815 | 0.797 | 0.800 | 0.924 |

ROC: receiver operating characteristic, MPerceptron: multi-layered perceptron.

expression such as "not achieved," which was observed in formal representation sentence, does not occur in the same sentence. The third means that the co-occurrence event that was observed in formal representation has three elements in the current sentence as in (#{(serum potassium), (creatinine), (monitored)} = 3). The fourth means that the number of phrasal templates that were observed in <RECOMMEND> sentences is 3. In Figure 3, you can find 3 phrasal templates − [should + be + VBN], [at + least + NUMBER], [NUMBER + times + per + year]. The fifth means that there are two phrasal templates which were observed in the sentence tagged <ANALYSIS>.

We use the tagged corpora for training/testing and also for creating feature extractors (Figure 4). Here, sentence instances were used for observing sentence construction events whose sub-events, such as phrasal events or occurrence of lexical items, were counted and used to make an event table with statistics. Table 3 shows what the event table looks like. Five event tables play a role in registering structural event features for a certain class. They also function as a lexicon with which to count the sub-events in the training and testing time, which is looked up at runtime. Those sub-events that belong to a different sentential class but have exactly the same surface were counted in each event table for later use in determining the weight.

The first column contains sub-events and the numerators/denominators in the second column signify the number of occurrences of each sub-event in a class and the total number of sub-event occurrences in four classes, respectively. For example, the total number of sub-event "at least + CD" in the

Table 6. Combination of NB classifiers by AdaBoost.M1

| Training order | Method | Precision | Recall | F-measure | ROC area |
|---|---|---|---|---|---|
| 1st | NB | 0.825 | 0.774 | 0.784 | 0.906 |
| 2nd | NB | 0.825 | 0.774 | 0.784 | 0.926 |
| 3rd | NB | 0.825 | 0.774 | 0.784 | 0.926 |
| 4th | NB | 0.825 | 0.774 | 0.784 | 0.926 |
| 5th | NB | 0.825 | 0.774 | 0.784 | 0.926 |
| 6th | NB | 0.825 | 0.774 | 0.784 | 0.926 |

ROC: receiver operating characteristic, NB: Naïve Bayes.

<FRS> class is 10 and its weight is 0.125. The identical sub-event "at least + CD" in the sentences tagged as <RECOMMEND> class occurs 70 times, its weight being 0.875. These weights transform the input data at runtime into a weighted feature vector. For example, an input [3, 1, 2, 3, 2] shall be applied to the weight function if the data contains the sub-event "at least + CD" so that the first and the fourth values may get weighted to better reflect the data. $T_{weight}$ ([3, 1, 2, 3, 2]) = [0.375, 1, 2, 2.625, 2]

## III. Results

A 10-fold cross validation using 340 sentences was done. Although the typical method is to split data into training, testing and validation sets, n-fold cross validation is preferred when the available data are highly limited in size. So we split the data into 10 sub-sets and tested on every nth data after training each model using the rest of the data, calculating precision, recall and f-measure on average. The ratio of the size of sentences for four classes was identical, the number of instances thus being 85 per class. Each classifier was trained using a set of 340 feature vectors.

Much to the contrary to the initial intuition, the f-measure of overall classifiers given in Table 4 was over 0.78. The performance of Naïve Bayes was relatively lower than other algorithms, probably due to its being based on independence assumption, which does not stand in harmony with the non-linear nature inherent in textual object generation systems. Since this research presupposed that these base learners would choose weak classification model parameters and that it would be necessary for them to be sequentially combined by boosting for learning strong classifiers based on a weighted voting scheme, the result above simply eradicated the necessity of using a meta-algorithm (Table 5).

Since weak classifiers are generally defined as having slightly better accuracy than a random classifier with an accuracy of less than 50%, we concluded that even boosting Naïve Bayes classifiers would not cause distinct improvement. Table 6 is the actual test in which we sequentially combined Naïve Bayes classifiers using the AdaBoost.M1 meta-algorithm.

As given in (Table 6), the improvement of Naïve Bayes classifiers combined by AdaBoost.M1 was not sufficiently conspicuous. This is because the transformation of the original feature space by feature extractors into a lower dimension may have reduced the complexity and non-linearity of the data, decreasing the width of scatter and the number of outliers. In actual fact, boosting Neural Network algorithm such as Multi-layered Perceptron or Radial Basis Function Network saw no increase of accuracy by AdaBoost.M1.

## IV. Discussion

The aims of this research were to apply transformation to tackle the problem of dimensionality and to increase classification accuracy by applying a boosting algorithm to learn a strong classifier that is robust to outliers and the non-linear characteristics of data. The second purpose turns out to be meaningless when the curse of dimensionality is resolved and robust classification algorithms such as a multilayer perceptron or a radial basis function network are adopted.

Moreover, we found that transformation has the advantage of exploiting structural and underlying features which go unseen by the BOW model. We also realized that integrating a sentential classifier with a TF-IDF-based search engine enhances a search process by realizing the capability of maximizing the probability of automatically presenting relevant information required in the context generated in the guideline authoring environment. This, however, has a disadvantage of increasing the total amount of time required to parse and classify the set of sentences within a document at runtime. Therefore, our future study shall be focused on excluding slow parsing processes while extracting structural features from a textual object.


http://dx.doi.org/10.4258/hir.2011.17.4.224

## Conflict of Interest

No potential conflict of interest relevant to this article was reported.

## Acknowledgements

This research was supported by Grant No. 10037283 from the Industrial Strategic Technology Development Program funded by the Ministry of Knowledge Economy.

## References

1. Woolf SH, Grol R, Hutchinson A, Eccles M, Grimshaw J. Potential benefits, limitations, and harms of clinical guidelines. BMJ 1999; 318: 527-530.
2. Buchanan B, Shortliffe EH. Rule-based expert systems: the MYCIN experiments of the Stanford Heuristic Programming Project. Reading, MA: Addison-Wesley; 1984.
3. Pan F. Multi-dimensional fragment classification in biomedical text. Kingston, OT: Queen's University; 2006.
4. Xin L, Xuan-Jing H, Li-de W. Question classification by ensemble learning. Int J Comput Sci Netw Secur 2006; 6: 146-153.
5. Freund Y, Schapire RE. Experiments with a new boosting algorithm. In: Saitta L; European Coordinating Committee for Artificial Intelligence; Associazione italiana per l'intelligenza artificiale, eds. Thirteenth International Conference on Machine Learning. San Mateo, CA: Morgan Kaufmann Publishersn; 1996. p148-156.
6. Marsland S. Machine learning: an algorithmic perspective. Boca Raton, FL: Chapman & Hall/CRC; 2009.
7. Duda RO, Hart PE, Stork DG. Pattern classification. 2nd ed. New York: Wiley; 2000.
8. Salton G, McGill MJ. Introduction to modern information retrieval. New York: McGraw-Hill; 1983.
9. Domingos P, Pazzani M. On the optimality of the simple Bayesian classifier under zero-one loss. Mach Learn 1997; 29: 103-130.
10. Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other kernel-based learning methods. Cambridge, NY: Cambridge University Press; 2000.
11. Berger AL, Della Pietra SA, Della Pietra VJ. A maximum entropy approach to natural language processing. Comput Linguist 1996; 22: 39-71.
12. Darroch JN, Ratcliff D. Generalized iterative scaling for log-linear models. Ann Math Statist 1972; 43: 1470-1480.
13. Vapnik VN. Statistical learning theory. New York: Wiley Interscienc; 1998.
14. Ethem A. Introduction to machine learning. Cambridge, MA: MIT Press; 2004.
15. Cardoso-Cachopo A, Oliveira AL. An empirical comparison of text categorization methods. Lect Notes Comput Sci 2003; 2857; 183-196.
16. Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res 2003; 3: 1157-1182.
17. Yang Y, Pedersen JO. A comparative study on feature selection in text categorization. In: Proceedings of ICML-97, 14th International Conference on Machine Learning; 1997 Jul 8-12; Nashville, TN, USA. p412-420.
18. Feldman R, Sanger J. The text mining handbook: advanced approaches in analyzing unstructured data. Cambridge, NY: Cambridge University Press; 2007.
19. The Apache Software Foundation. Welcome to Apache Nutch [Internet]. The Apache Software Foundation; c2011 [cited at 2011 Sep 14]. Available from: http://nutch.apache.org/.
20. The Apache Software Foundation. Apache Tika: a content analysis toolkit [Internet]. The Apache Software Foundation; c2011 [cited at 2011 Sep 14]. Available from: http://tika.apache.org/.
21. OpenNLP. Welcome to Apache OpenNLP [Internet]. The Apache Software Foundation; c2010 [cited at 2011 Sep 14]. Available from: http://incubator.apache.org/opennlp/.
22. The Stanford Natural Language Processing Group (SNLP). The Stanford parser: a statistical parser [Internet]. The Stanford Natural Language Processing Group; [cited at 2011 Sep 14]. Available from: http://nlp.stanford.edu/software/lex-parser.shtml.
23. Shatkay H, Pan F, Rzhetsky A, Wilbur WJ. Multi-dimensional classification of biomedical text: toward automated, practical provision of high-utility text to diverse users. Bioinformatics 2008; 24: 2086-2093.
24. Chobanian AV, Bakris GL, Black HR, Cushman WC, Green LA, Izzo JL Jr, Jones DW, Materson BJ, Oparil S, Wright JT Jr, Roccella EJ; National Heart, Lung, and Blood Institute Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure; National High Blood Pressure Education Program Coordinating Committee. The Seventh Report of the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure: the JNC 7 report. JAMA 2003; 289: 2650-2672.