2012 10th IAPR International Workshop on Document Analysis Systems

Multiple Feature-Classifier Combination in Automated Text Classification

Lazaro S.P. Busagala[*], Wataru Ohyama[†], Tetsushi Wakabayashi[†] and Fumitaka Kimura[†]
[*]*Sokoine National Agricultural Library*
Sokoine University of Agriculture, Morogoro, Tanzania
Email: busagala@suanet.ac.tz
[†]*Graduate School of Engineering, Mie University, Japan*
Email: see http://www.hi.info.mie-u.ac.jp/en/index.html

***Abstract*—Automatic text classification (ATC) is important in applications such as indexing and organizing electronic documents in databases leading to enhancement of information access and retrieval. We propose a method which employs various types of feature sets and learning algorithms to improve classification effectiveness. Unlike the conventional methods of multi-classifier combination, the proposed method considers the contributions of various types of feature sets and classifiers. It can therefore be known as multiple feature-classifier combination (MFC) method. In this paper we present empirical evaluation of MFC using two benchmarks of text collections to determine its effectiveness. Empirical evaluation show that MFC consistently outperformed all compared methods.**

***Keywords*-Feature-Classifier Combination; Multi-classifier combination; ensembles; Text Classification/Categorization; Feature reduction**

I. Introduction

The process of automatically assigning new documents to pre-defined categories based on training examples is called automatic text classification (ATC). The increased availability of electronic documents creates the importance of having accurate methods in ATC. There are many applications of ATC which include spam filtering, information retrieval and web information categorization. One of the important step in ATC is feature generation and selection for document representation before feeding into a learning algorithm.

This paper proposes a method based on multiple feature-classifier combination (MFC). In the conventional methods of multiple classifier combination, only one type of feature is used.

Unlike the conventional way of classifier combination, in this technique, various types of features are separately fed to different classifiers. Then the decisions of classification algorithms are combined to improve the classification effectiveness. The classifier decisions were combined by the use of the majority vote function.

The rest of this paper is organized as follows. Section II describes the proposed method. Section III explains some implementation issues including the classification experiments to verify the effectiveness of the proposed framework. In section IV, we discuss the experimental results. Section V gives a survey of related works. Finally, a summary and future research possibilities are given in Section VI.

II. Multiple Feature-Classifier Combination (MFC)

This section describes techniques on which the proposed approach is based. We describe the features that are used in building the proposed model. Furthermore, the framework of our approach is illustrated and described. Many research work on multiple classifier combination (MCC) has been focusing on combination functions where by classifier decisions from one type of features are combined.

Unlike the conventional way of combining the classifier decisions, we propose use of multiple features and classifiers in combining decisions, thus the name multiple feature-classifier combination (MFC). The idea is that multiple features complement to each other such that classification errors can be reduced. Figure 1 illustrates the classification procedure that makes use of the algorithm with multiple feature-classifier combination. Step 1 and 2 in Figure 1 are described in Sections II-A, II-B and II-C. Feature reduction in step 3, refers to any suitable method such as principal component analysis and discriminant analysis.

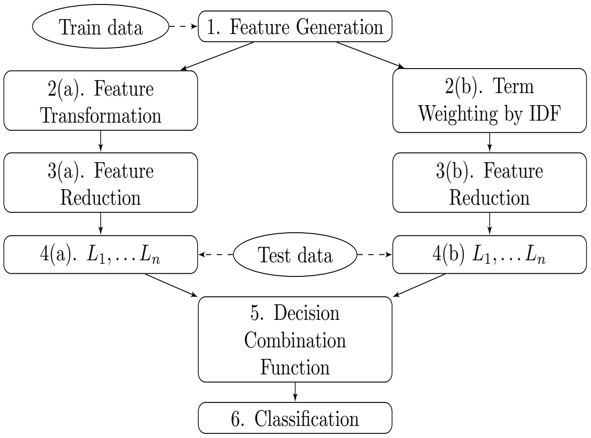

Figure 1. The automated text classification algorithm for multiple feature-classifier combination (MFC). Feature reduction methods can be applied including principal component analysis and discriminant analysis techniques. L_i refers to the learning methods for classification which are trained before the unseen data (test data) can enter the classification algorithm. IDF is the abbreviation for inverse document frequency.

978-0-7695-4661-2 2012

DOI 10.1109/DAS.2012.56

A. Feature Generation

Let us consider a set of N sample texts, $\chi = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ with n–dimensional text space. Let us assume that every textual document belongs to one of the C classes $\{\omega_1, \omega_2, \ldots, \omega_C\}$. Each text can be represented as a feature vector, $\mathbf{x}_k = [x_1 x_2 \ldots x_n]^T$, whereby x_i is the term frequency and T refers to the transpose of a vector. We follow this way of generating features and construct two types of features: (1) term frequency weighted by inverse document frequency (TFIDF) and (2) relative frequency with power transformation (RFPT).

B. Term Frequency Weighted with Inverse Document Frequency (TFIDF)

The components of the feature vectors are the term frequencies weighted by inverse document frequency (TFIDF). This technique has been borrowed from information retrieval (IR) [2], [3]. Formerly, the term weighting is defined as

$$w_i = x_i * log\frac{N}{N_i}, \tag{1}$$

where N_i is the *document frequency* which is the number of documents in which term i occurs. In other words the log part of the equation (1) denotes the *inverse document frequency* (IDF).

The intuition here is that a *term* which occurs in many documents is not a good discriminator for retrieving desired documents. Therefore it should be given less weight than the one which occurs in few documents [3], [4]. In order to avoid text length variation within documents, a normalization to vector unit length is carried out using

$$\acute{w}_i = \frac{w_i}{\sqrt{\sum_{j=1}^{n}(w_j)^2}}, \tag{2}$$

which is also called cosine normalization.

C. Relative Frequency with Power Transformation (RFPT)

Obtaining RFPT involves transforming absolute term frequency (AF) to relative term frequency (RF) and power transformation (PT). The resulting features can be called Relative Frequency with Power Transformation (RFPT). These are obtained by the expression:

$$z_i = \left(\frac{x_i}{\sum_{j=1}^{n} x_j}\right)^v, \quad (0 < v < 1\ ), \tag{3}$$

The advantages of RFPT include lack of dependency on text length. In addition, the shape of the sample distributions is Gaussian-like. Based on the optimality of the linear or quadratic classifiers which are designed for Gaussian distributions, this kind of transformation is of advantageous to text classification systems because of reduced mis-classifications.

Furthermore, it is also worth noting that the length of RFPT is normalized to 1 when $v = 0.5$ as follows:

$$\sum_{i=1}^{n} z_i^2 = \sum_{i=1}^{n}\left(\frac{x_i}{\sum_{j=1}^{n} x_j}\right) = 1. \tag{4}$$

In other words RFPT satisfies the normality property.

D. Feature Reduction

In feature (dimensionality) reduction, we propose an integrated discriminant analysis (IDA) which optimizes both variance ratio and the mean square error simultaneously. Therefore IDA can be regarded as the integrated optimization of Principal Component Analysis (PCA) and Canonical Discriminant Analysis (CDA) [5].

Let β be a constant in the range [0, 1]. Furthermore, let Λ and Φ denote eigenvalues and corresponding eigenvectors, respectively. Then, IDA can be treated as a generalized eigenvalue analysis of the form

$$(S_B + \beta S_W)\Phi = \{(1-\beta)S_W + \beta I\}\Phi\Lambda, \tag{5}$$

where S_B and S_W are between-class and within-class covariance matrices. I is the identity matrix.

When $\beta = 0$, expression 5 tends to be equivalent to CDA, and, when $\beta = 1$, it tends to be equivalent to the classical principal component analysis (PCA). IDA solves the following optimization problem:

$$\max \frac{\Phi^T(S_B + \beta S_W)\Phi}{\Phi^T\{(1-\beta)S_W + \beta I\}\Phi}. \tag{6}$$

The determination process of the integration parameter β can be estimated via cross-validation techniques as proposed in [6].

Using eigenvectors which correspond to $m (m \leq n)$ largest eigenvalues, discriminants $\mathbf{u} = [u_1 \ldots, u_m]$ are defined by the linear transformation

$$\mathbf{u} = \Phi^T \mathbf{x}. \tag{7}$$

The reduced dimensionality of features are composed of m discriminants $\mathbf{u} = [u_1 \ldots, u_m]$. This forms a projected or transformed data set $\Xi = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$ used in the classification process. IDA can extract enough number of features.

CDA has limitations including inability to extract more than $C-1$ features which might not be enough. Another limitation is that S_W is always singular when the dimensionality is greater than the number of training sample which is always the case in document classification. PCA also has limitations including the fact that it does not consider discriminatory properties of data points.

III. Experiments

A. Data for Experiments

We used two popular data sets in our experiments as described in the following two subsections.

1) Reuters-21578 Data Set: A benchmark collection for text categorization research called Reuters-21578 was used which has been widely employed by other researchers [2], [7]. Reuters-21578 is composed of 21,578 articles manually classified into 135 categories. One document may belong to one or more categories. Therefore Reuters-21578 poses both multi-class and multi-label problems.

We used the ModApte Split [7] which contains 12,902 articles. In this split the training set contains 9,603 documents and for the test set 3,299 documents, and 8,676 documents are not used. ModApte Split is the most commonly used split among the splits. A total of used 115 categories in the experiments were used.

2) OHSUMED(HD-119) Data Set: The OHSUMED collection was first published as text retrieval test collection in 1994 [8]. It contains 348,566 MEDLINE references from the years 1987 to 1991. Although all of the references have titles only 233,445 have abstracts. According to Lewis et al. [9], in text categorization problems queries and relevance judgments in the collection are ignored. We follow the split used in [9]. Categories are based on medical subject heading (MeSH categories).

The focus here was on 119 MeSH categories in the heart disease sub-tree (HD-119) of the cardiovascular diseases tree structure such as in [9]. In the experiments, after preprocessing and labeling, we randomly extracted 90 MeSH categories of HD-119. We use 12739 abstracts of documents as training data set. A total of 3742 abstracts of documents as test data set. Therefore results presented here are for 90 categories. HD-119 subtree is a multi-label problem meaning that one document may belong to one or more categories.

B. Lexicon Generation

In general, function words are not useful to represent document features discriminatory. Therefore, functional words and general words were removed with reference to a stop list from SMART[1] system described in [10]. This process reduces the features for the classification systems. This also reduces the amount of memory required for storage as well as processing time required by the classification systems.

Even when the stop list was used to remove useless words, many words remained. Thus, words with frequency value of 5 or less in all the training data were removed. The objective was to reduce further the remaining words. This removal of words, for example, reduced the lexicon from 24868 to 7474 terms in case of Reuters-21578. According to [2], this word removal does not affect the classification performance.

C. Classification and Performance Evaluation

Reasons for selecting a particular classifier to form an ensemble system include how best such a method performs. In the literature for automated text classification, k Nearest Neighbors and Support Vector Machines are among the best performers. Therefore they were selected in classification experiments to evaluate the proposed method.

1) k Nearest Neighbors (kNN): This method is one of the best performers among many that has been reported in the literature [2], [11]. The kNN algorithm relies on the concept that given a unseen document $\mathbf{x}$, the system finds the k nearest neighbors among the training documents to estimate its *a posteriori* probability $P(\omega_j|\mathbf{x})$ for each category [12].

Moreover, kNN can easily handle both multi-class and multi-label problems simultaneously as compared to other classification methods. Since the Reuters-21578 and OHSUMED collection are both multi-class and multi-label problems, thus kNN was used in the classification process.

2) Support Vector Machines (SVM): Support Vector Machines (SVMs) is the machine learning method which finds the optimal hyperplane with maximum margins. The nearest patterns to the decision boundaries are the support vectors.

We used the SVM^{Light2} package [13] in the experiments. We divide each classification task into C binary classification problems and adopt the one against the rest strategy. Kernel functions used include linear and polynomial type.

D. Experiments for Multiple Feature-Classifier Combination

In the experiments, the best three performers out of the classifiers were used. These are linear SVM, kNN and polynomial SVM. The feature-classifier combination was formed as follows: (1) Linear SVM's decisions from RFPT; (2) Polynomial SVM's decisions from RFPT; (3) Linear SVM's decisions from TFIDF; (4) kNN's decisions from RFPT; and (5) kNN's decisions from TFIDF. Figure 2 illustrates an example on how the experiments were conducted. Note that there is intuitively no impact of sequence of classifiers before combining in terms of performance. This is due to the fact that the combination function gives them the same weight in this case.

1) Performance Measures: We adopt the recall, precision and F-measure for performance evaluation of classification effectiveness. These measures are regarded as standard evaluation methods for classification system in automatic text classification. The definitions of these measures can be found in [2], [12]. Micro-averaging and macro-averaging strategies are usually adopted. For comparability with other previous works in the literature we adopt micro-averaging and report F-measure scores.

E. Statistical Analysis of Improvements

Statistical significance testing gives an insight into any apparent improvements in the performance of algorithms or methods. It is therefore desirable to perform statistical

[1] The list is found at ftp://ftp.cs.cornell.edu/pub/smart/english.stop

[2] This package can be freely obtained at http://svmlight.joachims.org/. We are grateful to acknowledge Thorsten Joachims for availing the software and his support.

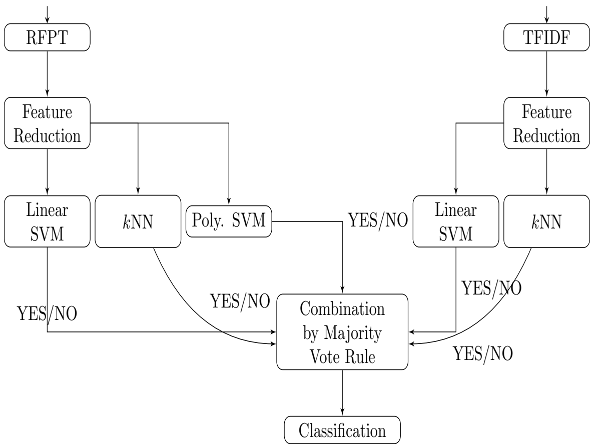

Figure 2. Example of the experiments with multiple feature-classifier combination (MFC). Features include relative frequency with power transformation (RFPT) and term frequency weighted by inverse document frequency (TFIDF).

analysis to show whether the proposed methods really have an effect on the performance of text categorization.

In the field of machine learning and other related fields, McNemar's test has been recommended to be one of the powerful statistical tests [14], [15]. Powerful in the sense that it has low probability of making type 1 error. Indeed various authors on text categorization have applied it successfully [16], [17].

IV. Empirical Results

This section discusses the results of the experiments. Table I summarizes the empirical results from two data sets. We provide results from conventional approaches such as of single type of features with single classifier and denote as SS. In this Table SS$_{(RFPT)}$ indicates the results of the best classifier when using RFPT which was Linear SVM. Similarly SS$_{(TFIDF)}$ are results from TFIDF features using Linear SVM which was the best performer.

MCC results were obtained by combining classifier decisions using majority voting rule. In this case the best features were used as conventionally done as shown by the literature. For this case, RFPT was the best. The classifiers combined are Linear SVM, Polynomial SVM and kNN.

Table I

Comparison of the micro-averaged F_1 scores (%). Features include relative frequency with power transformation (RFPT) and term frequency weighted by inverse document frequency (TFIDF). SS = refers to single best classifier i.e. Linear SVM, on respective feature type. MFC refers to multiple feature-classifier combination. Classifiers include linear SVM, polynomial SVM and kNN.

Data Set	SS$_{(RFPT)}$	SS$_{(TFIDF)}$	MFC	MCC
Reuters	88.53	88.35	**89.3**	89
OHSUMED	72.3	71.6	**73.7**	73

It is interesting to note that MFC consistently outperformed all other methods on all data sets used. In other words, in the case of the Reuters and OHSUMED data sets, MFC outperformed all other methods by achieving the highest micro-averaged $F_1 = 89.3\%$ and $F_1 = 73.7\%$, respectively. Considering these data sets and the respective splits, these could be the highest performance scores ever reported in the ATC literature. The results from the proposed method are statistically better than the conventional methods.

V. Related works

In the literature, most ATC works on building ensembles consider various classifiers with isolated single type of features. In other words conventionally the concept of ensembles does not associate the idea of type features used. They usually focus on the functions to combine decisions while using solely one type of features. This is the biggest difference with other works on the ensembles (a.k.a classifier committee).

Unlike our work in this paper, various researchers have worked on ensembles which most of them focused on combination functions [1], [19], [20], [21], [22], [23]. None of these works use a similar framework like what we are proposing in this paper.

Our approach allows any combination function of heterogeneous classifier to be implemented. This flexibility is illustrated in Figure 1. Secondly, the classification performance presented in our work are higher than what is published in [1] and other literature showing that if ensembles are constructed using multiple features and classifiers the classification effectiveness of learning algorithms can be enhanced. Furthermore, our approach simultaneously takes care of the problems of dependency on length; sample distribution (see Sections II); and the curse of dimensionality.

VI. Summary and Future Work

This paper proposes an approach called multiple feature-classifier combination (MFC) which improves text classification performance. Unlike conventional methods of multiple classifier combination, it employs various types of feature sets and classifiers to improve classification effectiveness.

The proposed multiple feature-classifier combination also improved the classification performance outperforming all the compared methods.

Potential future research includes use of multi-classifier combination using other combination functions and use of more samples. In addition, it may be of interest if this technique could be experimentally studied further in applications such as spam filtering and automated survey coding.

References

[1] G. P. C. Fung, J. X. Yu, H. Wang, D. W. Cheung, and H. Liu, "A balanced ensemble approach to weighting classifiers for text classification," in *ICDM '06: Proceedings of the Sixth International Conference on Data Mining*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 869–873.

[2] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[3] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, pp. 11–12, 1972.

[4] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for IDF," *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[6] J. H. Friedman, "Regularized discriminant analysis," *Journal of American Statistical Association*, vol. 84, no. 405, pp. 165–175, 1989.

[7] T. Zhang and F. Oles, "Text categorization based on regularized linear classification methods," *Information Retrieval Journal*, vol. 4, pp. 5–31, 2001.

[8] W. Hersh, "Ohsumed: an interactive retrieval evaluation and new large test collection for research," in *Proceedings of 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, 1994, pp. 192–201.

[9] D. Lewis, R. E. Schapire, J. Callan, and R. Papka, "Training algorithms for linear text classifiers," in *Proceedings of* 19^{th} *Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, 1996, pp. 298–306.

[10] G. Salton, *The SMART Retrieval System—Experiments in Automatic Document Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1971.

[11] J. Rennie, L. Shih, J. Teevan, and D. Karger, "Tackling the poor assumptions of naive bayes text classifiers," in *Proceedings of the 12th International Conference on Machine Learning (ICML)*, Washington DC, 2003), pp. 616–623.

[12] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the Twenty-First International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, pp. 42–49.

[13] T. Joachims, *Learning to classify text using support vector machines: Methods, Theory and Algorithms*. Kluwer Academic Publishers Boston Dordrecht London, 2001.

[14] T. G. Dietterich, "Approximate statistical test for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[15] L. Gillick and S. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proceeding of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1989, pp. 532– 535, glasgow.

[16] C.-M. Tan, Y.-F. Wang, and C.-D. Lee, "The use of bigrams to enhance text categorization," *Information Processing and Management: an International Journal*, vol. 38, no. 4, pp. 529–546, 2002.

[17] T.-Y. Wang and H.-M. Chiang, "Fuzzy support vector machine for multi-class text categorization," *Inf. Process. Manage.*, vol. 43, no. 4, pp. 914–929, 2007.

[18] P. Soucy and G. Mineau, "Beyond TFIDF weighting for text categorization in the vector space model," in *Proceeding of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2005, pp. 1130–1135.

[19] G. Giacinto and F. Roli, "Adaptive selection of image classifiers," in *Electronics Letters*. Springer Verlag Ed, 1997, pp. 38–45.

[20] L. Larkey and W. B. Croft, "Combining classifiers in text categorization." ACM Press, 1996, pp. 289–297.

[21] R. Liere and P. Tadepalli, "Active learning with committees for text categorization," in *In proceedings of the Fourteenth National Conference on Artificial Intelligence*, 1997, pp. 591–596.

[22] R. E. Schapire and Y. Singer, "Boostexter: A boosting-based system for text categorization," in *Machine Learning*, 2000, pp. 135–168.

[23] S. M. Weiss, C. Apte, F. J. Damerau, D. E. Johnson, F. J. Oles, T. Goetz, and T. Hampp, "Maximizing text-mining performance," *IEEE Intelligent Systems*, vol. 14, no. 4, pp. 63–69, 1999.