An Automated Advanced Clustering Algorithm for Text Classification

G. Narasimha Rao, R. Ramesh, D. Rajesh, D. Chandra Sekhar
1,2,4 Dept. of CSE, KITS, Vinjanampadu, Guntur, AP, India
3 Dept. of Computer Science and Engineering, NIT, JAMSHEDPUR, India

Abstract
A major characteristic of text document classification problem is extremely high dimensionality of text data. In this paper we present four new algorithms for feature/word selection for the purpose of text classification. We use sequential forward selection methods based on improved mutual information criterion functions. The performance of the proposed evaluation functions compared to the information gain which evaluate features individually is discussed. We present experimental results using naive Bayes classifier based on multinomial model, linear support vector machine and k-nearest neighbor classifiers on the Reuters data set.

Feature clustering is a powerful method to reduce the dimensionality of feature vectors for text classification. In this paper, we propose a fuzzy similarity-based self-constructing algorithm for feature clustering. The words in the feature vector of a document set are grouped into clusters, based on similarity test. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation. When all the words have been fed in, a desired number of clusters are formed automatically. We then have one extracted feature for each cluster. The extracted feature corresponding to a cluster is a weighted combination of the words contained in the cluster. Three ways of weighting, hard, soft, and mixed, are introduced. By this algorithm, the derived membership functions match closely with and describe properly the real distribution of the training data. Besides, the user need not specify the number of extracted features in advance, and trial-and-error for determining the appropriate number of extracted features can then be avoided. Experiments on real-world data sets show that our method can run faster and obtain better extracted features than other methods.

Keywords
Feature/word selection Support vector machine, Text Classification Fuzzy similarity, feature clustering, feature extraction,

I. Introduction
The goal of text document classification is to assign automatically a new document into one or more predefined classes based on its contents. An increasing number of statistical classification methods and machine learning algorithms have been explored to build automatically a classifier by learning from previously labelled documents including naive Bayes, k-nearest neighbor, support vector machines, neural network, decision trees, logistic regression. The overview of discusses the main approaches to text classification.

We propose a fuzzy similarity-based self-constructing feature clustering algorithm, which is an incremental feature clustering approach to reduce the number of features for the text classification task. The words in the feature vector of a document set are represented as distributions and processed one after another. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation. If a word is not similar to any existing cluster, a new cluster is created for this word. Similarity between a word and a cluster is defined by considering both the mean and the variance of the cluster. When all the words have been fed in, a desired number of clusters are formed automatically. We then have one extracted feature for each cluster. The extracted feature corresponding to a cluster is a weighted combination of the words contained in the cluster.

II. Definitions and Notations
Let \( C = \{ c_1, \ldots, c|C| \} \) be the set of \(|C|\) predefined classes and \( D = \{ d_1, \ldots, d|D| \} \) be the finite training document set. Let \( V = \{ w_1, \ldots, w|V| \} \) be the vocabulary set containing \(|V|\) distinct words occurred in training documents. Given a set of document vectors \( \{ d_1, \ldots, d|D| \} \) and their associated class labels \( c(d_k) = \{ c_1, \ldots, c|C| \} \), text classification is the problem of estimating the true class label of a new document. Text documents cannot be directly interpreted by a classifier. According to the bag-of-words representation, the document \( d \) can be represented by a feature vector consisting of one feature variable for each word in the given vocabulary set \( V \). A common characteristic of text data is its extremely high dimensionality. The number of potential features (several tens of thousands) exceeds the number of training documents.

To process documents, the bag-of-words model is commonly used. Let \( D = \{ d_1, d_2, \ldots, d_n \} \) be a document set of \( n \) documents, where \( d_1, d_2, \ldots, d_n \) are individual documents, and each document belongs to one of the classes in the set \( \{ c_1, c_2, \ldots, c|C| \} \). If a document belongs to two or more classes, then two or more copies of the document with different classes are included in \( D \).

Let the word set \( W = \{ w_1, w_2, \ldots, w_m \} \) be the feature vector of the document set. Each document \( d_i \), \( 1 \leq i \leq n \), is represented as \( d_i = <d_{i1}; d_{i2}; \ldots; d_{im}> \), where each \( d_{ij} \) denotes the number of occurrences of \( w_j \) in the \( i \)th document. The feature reduction task is to find a new word set \( W' = \{ w'_1; w'_2; \ldots; w'_{k< m} \} \), \( k \ll m \), such that \( W \) and \( W' \) work equally well for all the desired properties with \( D \). After feature reduction, each document \( d_i \) is converted into a new representation \( d'_i = <d'_{i1}; d'_{i2}; \ldots; d'_{ik}> \) and the converted document set is \( D' = \{ d'_1; d'_2; \ldots; d'_{n} \} \). If \( k \) is much smaller than \( m \), computation cost with subsequent operations on \( D' \) can be drastically reduced.

III. Dimensionality Reduction
Dimensionality reduction is a very important step in text classification because irrelevant and redundant features often degrade the performance of classification algorithms both in speed and classification accuracy. The number of features can be dramatically reduced by the domain dependent methods which include the elimination of stop words, stripping of special characters as well as stemming algorithms or morphological...
analysis. For a further dimensionality the domain independent methods can be used.

\[ S_0 = \sum_{c=1}^{k} P(c_q)(M_q - M_{c0})(M_q - M_{c0})^T \]

with \( P(c_q) \) being the prior probability for a pattern belonging to class \( c_q, M_q \) being the mean vector of class \( c_q, \) and \( M_0 \) being the mean vector of all patterns.

**D. Text Classification**

Text classification [10], (also known as text categorization or topic spotting) is the task of automatically sorting a set of documents into categories from a predefined set. This task has several applications, including automated indexing of scientific articles according to predefined thesauri of technical terms, filing patents into patent directories, selective dissemination of information to information consumers, automated population of hierarchical catalogues of Web resources, spam filtering, identification of document genre, authorship attribution, survey coding, and even automated essay grading. Automated text classification is attractive because it frees organizations from the need of manually organizing document bases, which can be too expensive, or simply not feasible given the time constraints of the application or the number of documents involved. The accuracy of modern text classification systems rivals that of trained human professionals, thanks to a combination of Information Retrieval (IR) technology and Machine Learning (ML) technology.

**E. Feature Retrieval**

Feature clustering is an efficient approach for feature reduction, which groups all features into some clusters, where features in a cluster are similar to each other. The feature clustering methods [are “hard” clustering methods, where each word of the original features belongs to exactly one word cluster. Therefore each word contributes to the synthesis of only one new feature. Each new feature is obtained by summing up the words belonging to one cluster. Let \( D \) be the matrix consisting of all the original documents with \( m \) features and \( D' \) be the matrix consisting of the converted documents with new \( k \) features. The new feature set \( W' = \{w'_1, \ldots, w'_k\} \) is obtained, which is a subset of the original feature set \( W. \) Then \( W' \) is used as inputs for classification tasks. Information Gain (IG) is frequently employed in the feature selection approach [10]. It measures the reduced uncertainty by an information-theoretic measure and gives each word a weight. The weight of a word \( w_j \) is calculated as follows:

\[
\begin{align*}
\text{IG}(w_j) &= \sum_{i=1}^{m} \left( P(c_i|w_j) \log \frac{P(c_i)}{P(c_i|w_j)} + P(w_j) \sum_{i=1}^{m} P(c_i|w_j) \log \frac{P(c_i|w_j)}{P(c_i)} \right) \\
&= \sum_{i=1}^{m} P(c_i|w_j) \log \frac{P(c_i)}{P(c_i|w_j)} + P(w_j) \sum_{i=1}^{m} P(c_i|w_j) \log \frac{P(c_i|w_j)}{P(c_i)}
\end{align*}
\]

where \( P(c_i|w_j) \) denotes the prior probability for class \( c_i, P(w_j) \) denotes the prior probability for feature \( w_i, P(w_j) \) is identical to \( 1 - P(w_j), \) and \( P(c_i|w_j) \) and \( P(c_i|w_j) \) denote the probability for class \( c_i \) with the presence and absence, respectively, of \( w_i. \) The words of top \( k \) weights in \( W \) are selected as the features in \( W'.\)

In feature extraction approaches, extracted features are obtained by a projecting process through algebraic transformations. An Incremental Orthogonal Centroid (IOC) algorithm was proposed in [14]. Let a corpus of documents be represented as an \( m \times n \) matrix \( X \in \mathbb{R}^{m \times n}, \) where \( m \) is the number of features in the feature set and \( n \) is the number of documents in the document set. IOC tries to find an optimal transformation matrix \( F^* \in \mathbb{R}^{n \times k}, \) where \( k \) is the desired number of extracted features, according to the following criterion:

\[
F^* = \arg \max \text{ trace}(FSF),
\]

where \( F^* \in \mathbb{R}^{n \times k} \) and \( FT = I, \) and
Therefore, we have m word patterns in total. For example, suppose we have four documents \(d_1, d_2, d_3, \text{ and } d_4\) belonging to \(c_1, c_2, c_3, \text{ and } c_4\), respectively. Let the occurrences of \(w_i\) in these documents be 1, 2, 3, and 4, respectively. Then, the word pattern \(x_i\) of \(w_i\) is:

\[
P(c_1 | w_1) = \frac{1 \times 1 + 2 \times 1 + 3 \times 0 + 4 \times 0}{1 + 2 + 3 + 4} = 0.3
\]

\[
P(c_2 | w_2) = \frac{1 \times 0 + 2 \times 0 + 3 \times 1 + 4 \times 1}{1 + 2 + 3 + 4} = 0.7
\]

(10)

Let \(X_i = \langle 0.3; 0.7 \rangle\). We consider the global filtering approach to feature selection in text document task. In this section novel algorithms for feature selection using mutual information are presented.

1. Feature Selection using Mutual Information

Our feature subset selection problem is formulated as follows: Given an initial set \(V\) with \(|V|\) features, find the subset \(S\) \(\subseteq V\) with \(|S|\) features that maximizes the mutual information for text defined as mutual information for a set of words averaged over all classes given by the following formula:

\[
MI(S) = \frac{\sum_{c \in \mathcal{C}} I(c, S)}{|S|}
\]

The mutual information for a feature/word \(w\) (word \(w\) occurred) averaged over all classes is defined as:

\[
MI(w) = \frac{\sum_{c \in \mathcal{C}} P(c) I(c, w)}{|S|}
\]

Here \(P(w)\) is the probability, that the word \(w\) occurred, \(P(c)\) is the probability of the class \(c\), \(P(w|c)\) is the conditional probability of the word \(w\) given the class \(c\), \(I(c, w)\) is the mutual information between class \(c\) and word \(w\). We can consider three strategies for solving our feature selection problem. The optimal strategy generates all the word subsets \(S\) and compares their \(MI(S)\). It is almost impossible for too many combinations. In the backward elimination strategy we remove the worst word from the complete set \(S\) one by one till the required number of words remain. This procedure has a lot of difficulties in computing \(I(c, S)\).

IV. Clustering Algorithms

The whole clustering algorithm can be summarized below.

Initialization

- \# of original word patterns: \(m\)
- \# of classes: \(p\)

Threshold: Initial deviation: \(\rho\)
Initial deviation: \(c_0\)
Initial of clusters: \(k = 0\)

Input:

- \(x_i = \langle x_{i1}, x_{i2}, \ldots, x_{ip} \rangle, 1 \leq i \leq m\)

Output:

- Clusters \(G_1, G_2, \ldots, G_k\)

procedure Self-Constructing-Clustering-Algorithm for each word pattern \(x_i, 1 \leq i \leq m\)

\[
\text{temp}_W = G_1 \cup G_i (x_i); 1 \leq j \leq k; \text{if } (\text{temp}_W = \emptyset) \text{ A new cluster } G_j, h = k + 1\]

else let \(G_t \in \text{temp}_W\) be the cluster to which \(x_i\) is closest by (19);

Incorporate \(x_i\) into \(G_t\) by (20)-(24);

endif;

endif;

return with the created \(k\) clusters;

endprocedure

Note that the word patterns in a cluster have a high degree of similarity to each other. Besides, when new training patterns are considered, the existing clusters can be adjusted or new clusters are added.
can be created, without the necessity of generating the whole set of clusters from the scratch. The order in which the word patterns are fed in influences the clusters obtained. We apply a heuristic to determine the order. We sort all the patterns, in decreasing order, by their largest components. Then the word patterns are fed in this order. In this way, more significant patterns will be fed in first and likely become the core of the underlying cluster. For example, let x₁ = <0:1; 0:3; 0:2; 0:4; 0:5>.

Fig. 4: Feature Clustering Diagram

V. Conclusions
This paper attacked the problem of feature selection for document classification. In this paper we presented methods based on novel improved mutual information measures. The proposed algorithms are new in the text classification field. It uses good optimization performance of support vector machines to improve classification performance of genetic algorithm with elite strategy. Iris dataset and a text dataset are chosen to validate performance of the combing algorithm. It’s obviously that the hybrid algorithm and future Clustering can be applied to classify literatures in the field of electrical engineering. Future study direction will focus on the effect to performance when related parameters, such as crossing-over rate, mutation rate, size of population, etc., have different values, and improve computational efficiency of the new algorithm further.

Similarity-based clustering is one of the techniques we have developed in our machine learning research. In this paper, we apply this clustering technique to text categorization problems. We are also applying it to other problems, such as image segmentation, data sampling, fuzzy modeling, web mining, etc. We found that when a document set is transformed to a collection of word patterns, the relevance among word patterns can be measured, and the word patterns can be grouped by applying our similarity-based clustering algorithm. Our method is good for text categorization problems due to the suitability of the distributional word clustering concept.

References