Hybrid supervised clustering based ensemble scheme for text classification

Abstract

Purpose- The immense quantity of available unstructured text documents serve as one of the largest source of information. Text classification can be an essential task for many purposes in information retrieval, such as document organization, text filtering and sentiment analysis. Ensemble learning has been extensively studied to construct efficient text classification schemes with higher predictive performance and generalization ability. Providing diversity among the classification algorithms is a key issue in the ensemble design.

Design/methodology/approach- An ensemble scheme based on hybrid supervised clustering is presented for text classification. In the presented scheme, supervised hybrid clustering, which is based on cuckoo search algorithm and k-means, is introduced to partition the data samples of each class into clusters so that training subsets with higher diversities can be provided. Each classifier is trained on the diversified training subsets and the predictions of individual classifiers are combined by the majority voting rule. The predictive performance of the proposed classifier ensemble is compared to conventional classification algorithms (such as Naïve Bayes, logistic regression, support vector machines and C4.5 algorithm) and ensemble learning methods (such as AdaBoost, Bagging and Random Subspace) using eleven text benchmarks.

Finding- The experimental results indicate that the presented classifier ensemble outperforms the conventional classification algorithms and ensemble learning methods for text classification.

Originality/value- The presented ensemble scheme is the first to use supervised clustering to obtain diverse ensemble for text classification

Keywords: Text classification, classifier ensemble, diversity, supervised clustering

Paper type: Research paper

1. Introduction

Text classification (also known as text categorization) is the process of assigning text documents into one or more predefined categories based on the content of those documents. Given the immense quantity of unstructured text documents available, text classification becomes an essential task in machine learning and information retrieval for researchers and practitioners. Text classification is of great importance in several fields, such as automatic indexing, document organization, text filtering, hierarchical categorization of Web pages, word sense disambiguation, automated survey coding, authorship attribution, genre classification and spam filtering (Sebastiani, 2005).

Machine learning techniques have been widely employed for text classification. Ensemble learning is a research direction in machine learning, which aims to combine the predictions of multiple learning algorithms, so that a classification model with higher generalization ability and predictive performance can be achieved. Ensemble learning reduces the variance and bias of learning algorithms. In addition, the dependency of the classification to the characteristics of a single training set is eliminated (Kuncheva, 2005). Ensemble learning methods are

expected to outperform the base learning algorithms, owing to their statistical, representational and computational characteristics (Ditterich, 2000). Regarding the performance of ensemble schemes, obtaining base learning algorithms with higher predictive accuracy and diversity are very critical issues (Zhou, 2012). In order to achieve high diversity among the base learning algorithms, data-level or model-generation level manipulation can be performed (Mendes-Moreira et al., 2012).

In this paper, we present an ensemble classification scheme based on hybrid supervised clustering for text categorization. In the presented scheme, supervised clustering is employed to obtain diverse training subsets from the training set. Since supervised clustering algorithms tend to partition the examples from the same class into the same cluster, training sets constructed by the pairwise combination of different clusters are expected to be diverse (Xiao et al., 2016). In this scheme, supervised clustering is first employed to partition the instances of training set into the clusters. Then, clusters from different classes are pair wisely combined to obtain the training subsets for each base learning algorithm. In the supervised clustering stage, a hybrid supervised clustering algorithm based on cuckoo search algorithm and k-means algorithm is presented.

The main contributions of the paper can be summarized as follows:

- We present a novel, efficient ensemble classification scheme based on hybrid supervised clustering. In this scheme, supervised clustering algorithm is employed to obtain an efficient classifier ensemble by achieving high diversity.
- The paper presents a novel supervised clustering scheme which combines supervised k-means algorithm with the cuckoo search algorithm. In the hybrid clustering scheme, the cuckoo search algorithm is utilized to optimize weight values of supervised k-means algorithm.
- To the best of our knowledge, the presented ensemble scheme is the first to use supervised clustering to obtain diverse ensemble for text classification. The presented hybrid clustering algorithm, which integrates the cuckoo search algorithm and supervised k-means algorithm, is also a novel contribution to supervised clustering.

The rest of this paper is structured as follows. In Section 2, the state of the art in ensemble learning methods based on clustering and ensemble learning methods on text classification are presented. In Section 3, the theoretical foundations are briefly described. In Section 4, proposed classification scheme based on supervised clustering is presented. Section 5 presents the experimental analysis of the study. Finally, Section 6 presents the concluding remarks of the study.

2. Related work

This section briefly reviews related work on ensemble learning methods based on clustering and ensemble learning methods on text classification.

2.1. Ensemble learning methods based on clustering

The clustering algorithms can be included to the classification process to improve the predictive performance of classifier ensembles. For instance, Lee et al. (2010) presented an ensemble classification scheme based on Random forest algorithm and clustering for lung nodule classification. In the presented scheme, clustering was employed to partition all instances of training data into clusters so that the similarity among features of different classes

can be explored. In another study, Zhang and Lu (2010) presented an ensemble classification scheme based on fuzzy clustering with deflection. In this scheme, fuzzy c-means clustering algorithm was employed to partition the training data to obtain the distribution of the training set. Tsai (2014) presented a hybrid classification scheme for financial distress prediction. In the presented scheme, classifier ensembles were trained by the clustering results, obtained by self-organizing maps and k-means clustering. Haghighi et al. (2012) presented a weight tuning based approach to construct classifier ensembles with high diversity. In another study, Lin et al. (2014) proposed a hybrid ensemble classification scheme based on k-means clustering to identify an optimal subset of classifiers to be included in the classifier ensemble. Similarly, Zhang and Cao (2014) utilized clustering algorithm in the classifier ensemble to identify a set of classifiers from the ensemble. In another study, Rahman and Verma (2013) proposed a classifier ensemble based on clustering and genetic algorithm. Similarly, Farahbod and Eftekhari (2013) utilized subtractive clustering algorithm to extract fuzzy classification rules from data. In this scheme, gradient descent algorithm was employed to optimize the rule parameters. More recently, Meng et al. (2016) presented a classifier ensemble selection scheme based on affinity propagation clustering. In another study, Cardoso et al. (2016) presented a clustering based weightless neural network classifier ensemble for financial credit analysis. Similarly, Cruz et al. (2016) presented a radial basis function neural networks based classification scheme, which utilizes bee inspired data clustering algorithm for automatically identifying the number, location and dispersions of basis functions of radial basis function neural network classifiers. In another study, Song et al. (2016) presented an ensemble classification scheme for textual stream mining, where clustering trees were chosen based on an adaptive strategy. In another study, Xia and Zhang (2016) presented an ensemble classification scheme for credit scoring based on supervised k-means clustering algorithm. In this scheme, supervised clustering was employed to partition the dataset into clusters so that instances with the same class labels can be deployed at the same classes.

2.2. Ensemble learning on text classification

Machine learning methods have been successfully employed in text classification. Ensemble learning methods have been widely utilized to enhance the predictive performance of machine learning classifiers on text classification. For instance, Shi et al. (2011) presented a semi-supervised classification scheme for text categorization based on tolerance rough set and ensemble learning. In this scheme, approximate concepts of text documents were identified with the use of rough set theory. In the classification phase, an ensemble method based on support vector machines and Naïve Bayes algorithm was utilized. In another study, Yang et al. (2011) proposed an ensemble classification scheme for text data stream classification, where keywords were utilized to represent text documents. In another study, Xia et al. (2011) examined the contribution of ensemble learning on text sentiment classification. In the empirical analysis, predictive performance of different feature sets (such as part of speech based features and word-relation based features) were evaluated in conjunction with classification algorithms (such as Naïve Bayes, maximum entropy and support vector machines) and ensemble learning methods (such as fixed combination, weighted combination and meta-classifier combination). Similarly, Wang et al. (2014) examined the predictive performance of three ensemble learning methods (namely, Bagging, Boosting and Random Subspace) in conjunction with five classification algorithms (namely, Naïve Bayes, maximum entropy, decision tree, k-nearest neighbor and support vector machines) for text sentiment classification. In another study, an ensemble text classification scheme based on support vector machines, k-nearest neighbor algorithm and variable precision rough set theory was introduced (Li et al., 2011). More recently, Onan et al. (2016b) examined the predictive

performance of five statistical keyword extraction methods in conjunction with classification algorithms (such as Naïve Bayes, support vector machines, logistic regression and Random Forest) and ensemble learning methods (such as AdaBoost, Bagging, Dagging, Random Subspace and majority voting) for text classification. In another study, Onan et al. (2016c) presented an efficient ensemble scheme for text classification, where multi-objective differential evolution algorithm was utilized to assign optimal weight values to the classifiers of the ensemble. In another study, Elghazel et al. (2016) presented an ensemble classification scheme based on latent semantic indexing and document bootstrapping for multi-label text categorization. In this scheme, diversity of the ensemble was provided by random splits of the vocabulary. In another recent study, Lochter et al. (2016) introduced an ensemble scheme for detecting opinion in short text messages, based on text normalization, semantic indexing techniques and classification algorithms. Similarly, Perikos and Hatzilygeroudis (2016) presented an ensemble classification scheme based on statistical classifiers (such as Naïve Bayes and maximum entropy) and knowledge-based analysis of natural language sentences for recognizing emotions in text documents. In another study, Rodger (2015) presented an ensemble medical text document classification scheme based on K-means clustering and k-nearest neighbor algorithm to predict survival, mortality and morbidity rates of the patients. More recently, Al-Salemi et al. (2016) presented an enhanced text classification scheme based on Boosting round and labeled latent Dirichlet allocation and AdaBoost algorithm.

2.3. Motivation and contributions

As it can be observed from the earlier studies outlined in Sections 2.1 and 2.2, ensemble learning is an important paradigm of machine learning to enhance the predictive performance of classification algorithms. In order to obtain an efficient classifier ensemble, diversity among the base classifiers should be provided. There are different approaches to provide diversity on the ensemble, such as manipulating the input features and manipulating the classification algorithms. The main motivation of the study is to develop an efficient ensemble scheme for text classification by achieving high diversity among the classifiers. In this regard, we have adopted the idea of utilizing supervised clustering to provide diversified training subsets to the classification algorithms (Xiao et al., 2016). In the supervised clustering phase, a novel supervised clustering algorithm based on cuckoo search and k-means is introduced. To the best of our knowledge, it is the first study to utilize cuckoo search in conjunction with k-means algorithm for supervised clustering. In addition, it is the first study to use a supervised clustering based classifier ensemble for text classification.

3. Theoretical foundations

This section briefly explains the classification algorithms, ensemble learning methods and the latent Dirichlet allocation utilized in the proposed classification scheme.

3.1. Classification algorithms

3.1.1. Naïve Bayes algorithm

Naïve Bayes algorithm (NB) is a generative classifier based on Bayes' theorem. In text classification, NB algorithm employs a probabilistic model with independence assumptions to model the distribution of the text documents. In this scheme, the posterior probability of each class is computed based on the distributions of words in the text documents, while not taking the positions of the terms into account. In NB algorithm, the posterior probability of each

class is computed based on generative models and Bayes rule and each document is assigned to the class with the highest posterior probability (Aggarwal and Zhai, 2012). The assumption of conditional independence simplifies the required computations. Hence, the algorithm can scale well in application areas with high dimensional feature space, such as text mining. Although NB has a simple structure, the algorithm can achieve comparable results to other classification algorithms, such as decision trees and neural networks (John and Langley, 2005).

3.1.2. Support vector machines

Support vector machines (SVM) are linear classifiers that can be used for classification and regression analysis (Vapnik, 1995). Support vector machines aim to identify decision boundaries within the search space which can best separate one class from another class. SVM can perform well with high dimensional feature spaces, since it aims to determine an optimum direction of discrimination in the feature space. Text classification is a domain, which suffers from high dimensionality and sparsity of the feature space. In addition, features of text documents can be organized into linearly separable classes. Hence, support vector machines are suitable tools to classify text documents (Aggarwal and Zhai, 2012).

3.1.3. Logistic regression

Logistic regression (LR) is a linear classifier that models the probability of events' occurrence as a linear function of a set of predictor variables (Kantardzic, 2010). In logistic regression, the decision boundaries are identified based on a linear function of the features. Logistic regression classifier aims to optimize the likelihood function. Based on the likelihood function, the class label for documents can be identified. Logistic regression classifier can be trained by choosing parameters so that the conditional likelihood is maximized (Aggarwal and Zhai, 2012). Similar to support vector machines, logistic regression is also a linear classifier, which can yield promising results to text classification.

3.1.4. C4.5 decision tree classifier

C4.5 is a well-known decision tree classifier, which is a successor of ID algorithm (Quinlan, 1993). In C4.5 algorithm, information gain is employed as the test attribute selection criteria so that the attribute bias problem of ID3 can be eliminated. For a particular set, each time the algorithm selects an attribute with the highest information gain. The algorithm can work properly with continuous and default attribute values. Owing to its pruning mechanisms, the algorithm overcomes over-fitting, eliminates the exceptions and noise in the training set.

3.2. Ensemble learning methods

3.2.1. AdaBoost algorithm

The boosting algorithm is an ensemble learning method to convert weak learners, which are slightly better than random guessing, to strong learners, which can yield highly accurate rules or hypotheses for classification (Zheng and Xue, 2009). Boosting algorithm aims to obtain strong learners by training weak learning algorithms sequentially on different sampling distributions, where subsequent classifiers focus on the mistakes of former classifiers (Zhou, 2012). AdaBoost algorithm is one of the most influential boosting algorithm, which attempts to obtain a classification scheme with higher predictive performance by focusing on data

points that are difficult to classify (Freund and Schapire, 1996). In AdaBoost algorithm, all patterns of the training set are assigned initially to the same weight. The weight values for the patterns are adjusted by increasing the weight values for misclassified instances and by decreasing the weight values for correctly classified instances. In addition, weak classifiers have been assigned corresponding weight values. In AdaBoost algorithm, the sample distribution is adjusted so that some of the mistakes in the former iterations are eliminated (Zhou, 2012).

3.2.2. Bagging algorithm

Bagging (i.e., Bootstrap aggregating) algorithm is an ensemble learning method that employs bootstrap distribution to obtain different base learners (Breiman, 1996). In this scheme, diversity among base learning algorithms is achieved by obtaining training instances with sampling with replacement. The predictions of base learning algorithms trained on diversified training subsets are combined by majority voting or weighted majority voting. Bagging algorithm can yield promising results with datasets of limited size. In addition, it can perform predictive performance enhancement on unstable base learning algorithms.

3.2.3. Random subspace algorithm

Random subspace algorithm is an ensemble learning method that employs feature set manipulation to achieve diversity among base learning algorithms in the ensemble (Ho, 1998). In this scheme, base learning algorithms are trained on the randomly selected modified subspace features. The algorithm can avoid over-fitting and enhance the predictive performance. For datasets with many redundant features, Random subspace method can yield effective and efficient solutions (Zhou, 2012). Since text classification is a domain that is characterized by high dimensionality and irrelevancy of text features, Random Subspace method generally obtains promising results in text classification problems (Onan, 2016).

3.3. Latent Dirichlet allocation

The latent Dirichlet allocation model (LDA) is a widely employed generative probabilistic model to identify the latent topics in text documents (Blei et al., 2003). In LDA, each document is represented as a random mixture of latent topics and each topic is represented as a mixture of words. The mixture distributions are Dirichlet-distributed random variables to be inferred. In this scheme, each document exhibits the topics in different proportions, each word in each document is drawn among the topics and topics are chosen based on per-document distribution over topics (Blei, 2012). LDA attempts to determine the underlying latent topic structure based on the observed data. In LDA, the words of each document corresponds to the observed data. For each document in the corpus, words are obtained by following a two-staged procedure (Blei, 2012).

3.4. Supervised clustering

Supervised clustering is the process of partitioning instances based on a clustering algorithm with the aid of instances from a training set (Finley and Joachims, 2008). Supervised clustering aims to classify instances such that clusters have high probability density with respect to a single class. In addition, supervised clustering aims to keep the number of clusters low (Eick et al., 2004). Compared to the conventional clustering algorithms, the objective functions utilized for supervised clustering are different. In supervised clustering, class

impurity and number of clusters are taken into consideration. Class impurity corresponds to the percentage of minority instances in the different clusters of a clustering, whereas number of clusters should be kept as low as possible, while maximizing the class purity. In supervised clustering, instances are assigned into clusters based on the notion of closeness based on a particular fitness function. Since clusters have high probability density with respect to a single class, any instance that has been assigned to a particular cluster can be assumed to hold the same class label with the majority of the members of the cluster.

3.5. Cuckoo search algorithm

Cuckoo search algorithm is a metaheuristic optimization algorithm that is inspired by the aggressive reproduction strategies of cuckoo birds (Yang and Deb, 2009). Based on the reproduction strategy of cuckoos, mature cuckoos deploy their eggs in the nests of other host birds and species. The basic cuckoo search algorithm is based on three fundamental principles. First, each cuckoo lays one egg at a time and dump it in a randomly chosen nest. Secondly, the best nests with high quality of eggs will be carried over to the next generations. Thirdly, the number of available host nests is taken as a fixed number. In addition, the probability of discovering the laid egg by the host bird is $p_a \in [0, 1]$. Based on these three principles, the host bird can either remove the egg or abandon the nest and build a completely new nest. The general structure of the cuckoo search algorithm is outlined in Figure 1.

[Figure 1]

In cuckoo search algorithm, new solutions are determined based on the Levy flights. The Levy flights for generating new solutions $x_i^{(t+1)}$ is determined as given by Equation 1:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Levy(\lambda) \tag{1}$$

where $\alpha > 0$ denotes the step size associated to the scales of the problem, which is generally taken as $\alpha = 1$ and $\oplus$ corresponds to entry-wise multiplications. The Levy flight is a stochastic equation for a random walk and each random step length is drawn from Levy distribution.

Cuckoo search algorithm satisfies the global convergence properties (Wang et al., 2012). In addition, cuckoo search algorithm can explore the search space by local and global search. In this way, the search space can be explored in an efficient way and the identification of global optimal solutions can be guaranteed. Furthermore, Levy flights are employed in the identification of new solutions. Cuckoo search algorithm can explore the search space in a more efficient way compared to the other metaheuristics based on Gaussian processes (Yang and Deb, 2014). Hence, cuckoo search algorithm can be utilized as an efficient metaheuristic on several optimization problems (Gandomi et al., 2013; Yang and Deb, 2014).

4. Proposed classification scheme based on supervised clustering

K-means algorithm is one of the commonly used clustering algorithm to partition a dataset into *k* clusters (Jain, 2010). K-means algorithm works on a Euclidean space and assumes the number of clusters (*k*) is known in advance. In K-means algorithm, the centroid plays an essential role in assigning each object into the closest cluster. Given any set of points *C* in a metric space *M,* a point $c \in M$ can be defined as a centroid, which minimizes the criterion given in Equation 2 (Al-Harbi and Rayward-Smith, 2006):

$$\sum_{x \in C} \delta(c, x) \quad (2)$$

In K-means algorithm, k points are randomly selected to represent different clusters. These points are referred as the centroids of the clusters. For each of the points, the closest centroid to the particular point is determined. Thereon, each point is assigned to cluster of that centroid. In each iteration, the centers of clusters are updated by computing the mean values of the objects within the clusters. The process is repeated until clustering result does not change.

In clustering, a set of objects are assigned naturally to the groups such that the objects deployed in a particular cluster are more similar to each other than the objects of other groups. In contrast, supervised clustering aims to group objects based upon a priori hypotheses associated with the output of a clustering algorithm. In natural clustering, all objects have the same significance in determining the optimal partition of a dataset. However, supervised clustering aims to assign objects into different class labels based on a priori hypotheses. Supervised clustering involves a weighted metric to measure distances between objects, so that greater weight values are assigned to the objects with higher impacts on a particular class label. Hence, a weighted metric should be employed in a supervised clustering algorithm, such that some objects are assigned greater significance based on the significance of objects to the class labels.

Let R_+ be the set of positive real numbers that is R+= $\{x \in R \mid x \geq 0\}$, $M=R^n$ and $w \in R_+^n$ be a positive weight vector for $x, y \in M$, weighted Euclidean metric (δ_w) can be defined as given by Equation 3 (Al-Harbi and Rayward-Smith, 2006):

$$\delta_w(x, y) = \sqrt{\sum_{i=1}^{n} w_i (x_i - y_i)^2} \quad (3)$$

The identification of appropriate set of weights for k-means algorithm can be modelled as an optimization problem. In this regard, the hybrid supervised clustering algorithm that is employed in this paper aims to assign optimal weight values for supervised k-means algorithm with the use of cuckoo search algorithm. Hybrid supervised clustering algorithm initiates with assigning random weight values to each field. Then, k-means algorithm is run with the weighted Euclidean metric. Based on the weight values, the fitness of the weights is computed. Let cluster label assigned to a particular instance (*r)* be *C(r*) and let class label of that instance be *class(r),* the fitness of the weights is determined as given by Equation 4 (Al-Harbi and Rayward-Smith, 2006):

$$\left| \left(r \middle| class(r) = class(C_r) \right) \right| \quad (4)$$

Based on the notion of fitness of the weights, the weight values for each instance is optimized by running cuckoo search algorithm. The process is repeated for 30 iterations to obtain a set of appropriate weight values.

The proposed ensemble classification scheme utilizes hybrid supervised clustering algorithm based on supervised k-means and cuckoo search algorithms. Regarding the performance of ensemble learning methods, obtaining diverse base learning algorithms is an important issue. In this regard, the presented scheme seeks to achieve high diversity among the base learning algorithms of the ensemble by obtaining diversified training subsets. In order to do so, the paradigm of supervised clustering is employed to obtain diversified training subsets. In this way, diverse base learning algorithms are constructed. In addition, supervised clustering can

explore the spatial characteristics of the instances in each class, which can yield promising predictive performance on ensemble classification. Hence, base learning algorithms have also high local accuracy, in addition to high diversity (Vucetic and Obradovic, 2000; Xiao et al., 2016).

[Figure 2]

Since supervised clustering algorithms tend to partition the instances from the same class into the same cluster, supervised clustering can be initially employed on the full set so that each instance from different classes are grouped into different clusters. Let N denote the total number of distinct class labels in the original training set, training set can be divided into N different clusters with the help of supervised clustering algorithm. Since each cluster has instances with high probability density with respect to a single class, each training subset obtained at the end of the supervised clustering process is further divided into N clusters. Then, clusters from different classes are pair wisely combined to obtain the training subsets for each base learning algorithm. In the supervised clustering stage, a hybrid supervised clustering algorithm based on supervised k-means and cuckoo search is utilized, where cuckoo search is utilized to find optimal weights to maximize the fitness function defined in Equation 4. After obtaining the diversified training subsets, each base learning algorithm is trained on the newly constructed training subsets and the final prediction of the classifier ensemble is determined by the majority voting rule. The general structure of the proposed ensemble classification scheme is outlined in Figure 2.

In addition, text collections are generally characterized by high dimensional feature space. In order to represent text documents in a compact and efficient way, the Latent Dirichlet allocation method is utilized. In this scheme, each text collection have been modelled with the use of latent Dirichlet allocation and Gibbs sampling. In this representation, text documents may be represented by different number of latent topics. In order to obtain an appropriate value of topics for each document collection, we have considered different number of features ranging from 50 to 200 (i.e. 50, 100, 150 and 200). Since the highest predictive performance is achieved when 100 topics are used as features in each of the text collections, we adopted this representation scheme in our presented ensemble. Moreover, the identification of coherent topics is an important issue in latent Dirichlet allocation. To effectively filter incoherent (low quality) topics, we have utilized the topic coherence approach presented in Mimno et al. (2011).

5. Experimental results

In this section, we evaluate the predictive performance of the proposed classifier ensemble on text classification benchmarks. This section briefly presents the text classification datasets, evaluation measures, experimental procedure and the results of the experimental analysis.

5.1. Datasets

To evaluate the predictive performance of the proposed classification scheme, we have used eleven text benchmarks from several domains. In Table 1, the descriptive information regarding the text collections used in the empirical analysis is presented. The number of features listed in Table 1 corresponds to the number of terms extracted when vector space model is utilized to represent text documents (Rossi et al., 2013). In order to represent text documents, latent Dirichlet allocation and Gibbs sampling is employed. In this section, we provide results for a feature set of 100 topics. In order to pre-process data for topic modelling,

both stemming and stop word filtering was employed. In Stemming, Porter stemming was utilized to normalize all variations of words to a particular form. Furthermore, we removed stop words and words occurred only once.

[Table 1]

5.2. Evaluation metrics

To evaluate the predictive performance of proposed ensemble classification scheme, classification algorithms and well-known ensemble learning methods, classification accuracy and F-measure are utilized as the evaluation metrics.

Classification accuracy (ACC) is the proportion of true positives and true negatives over the total number of instances as computed by Equation 5:

$$ACC = \frac{TN+TP}{TP+FP+FN+TN} \tag{5}$$

where *TN, TP, FP* and *FN* represents number of true negatives, number of true positives, number of false positives and number of false negatives, respectively. Precision (*PRE*) is the proportion of the true positives against the true positives and false positives as given by Equation 6:

$$PRE = \frac{TP}{TP+FP} \tag{6}$$

Recall (*REC*) is the proportion of the true positives against the true positives and false negatives as given by Equation 7:

$$REC = \frac{TP}{TP+FN} \tag{7}$$

F-measure takes values between 0 and 1. It is the harmonic mean of precision and recall as computed by Equation 8:

$$F - measure = \frac{2*PRE*REC}{PRE+REC} \tag{8}$$

5.3. Experimental procedure

In the experimental analysis, 10-fold cross validation method is employed. For each fold, supervised clustering algorithms are reset, classifiers are retrained and the 100 topics are rediscovered. The experimental results reported in Section 5.4 indicate average results for 10-folds. The experimental analysis is performed with the machine learning toolkit WEKA (Waikato Environment for Knowledge Analysis) version 3.9 (Hall et al., 2009). The presented classification scheme is implemented in Java. We have also evaluated the performance of supervised k-means algorithm and hybrid supervised clustering algorithm (containing simulated annealing algorithm and supervised k-means algorithm) in the training subset manipulation phase (Al-Harbi and Rayward-Smith, 2006; Xiao et al., 2016). In Table 2, the parameter values used in the experimental analysis with metaheuristic-based supervised clustering schemes are presented.

[Table 2]

5.4. Results and discussion

To evaluate the predictive performance of the proposed classifier ensemble, we have used eleven text classification benchmarks from several different domains. In Tables 3–4, the best results achieved by a particular algorithm with the datasets appear in boldface, whereas the second results appear in bold italics. Table 3 presents average classification accuracies obtained with base learning algorithms, conventional ensemble learning methods and supervised clustering based classifier ensembles. Regarding the predictive performance of base learning algorithms, the greatest classification accuracies for all text classification datasets utilized in the empirical analysis were obtained with the support vector machines classifier. For classic4 dataset, support vector machines achieve a classification accuracy of 95.14%. The classification accuracies obtained by logistic regression and Naïve Bayes algorithm are also high for some datasets.

[Table 3]

As it can be observed from the results presented in Table 3, conventional ensemble learning methods outperform the base learning algorithms for text classification benchmarks. In addition, Table 3 also presents the experimental results obtained with three different supervised clustering based ensemble schemes. In supervised k-means based classifier ensemble (KM), supervised k-means algorithm was employed to manipulate training subsets. In contrast, simulated annealing and supervised k-means based classifier ensemble (SAKM) utilized a hybrid supervised clustering algorithm, where simulated annealing is utilized to adjust the weight values for supervised k-means. Finally, proposed hybrid supervised clustering based classifier ensemble (HSC) utilized a hybrid supervised clustering algorithm, where cuckoo search is utilized to adjust the weight values. The first concern of the study is to identify whether diversified training subsets based on supervised clustering can enhance the predictive performance on text classification. Compared to the conventional ensemble learning methods and base learning algorithms, the ensemble schemes achieved by supervised clustering based training subset manipulation yield more promising results on text classification. Supervised k-means algorithm based classifier ensemble, simulated annealing and supervised k-means based classifier ensemble and proposed hybrid supervised clustering based classifier ensemble obtain higher predictive performance in terms of classification accuracy compared to the other schemes. The second concern is to identify whether metaheuristic based weight adjustment can enhance the performance of supervised clustering in ensemble learning. In this regard, simulated annealing and cuckoo search algorithms are taken into consideration. As it can be observed from the results listed in Table 3, simulated annealing and supervised k-means based classifier ensemble and proposed hybrid supervised clustering based classifier ensemble yield more promising results compared to the supervised k-means algorithm based classifier ensemble. Hence, the performance obtained by supervised k-means based classifier ensemble can further be enhanced if metaheuristic based weight adjustment is employed. Regarding the predictive performance among all classification algorithms, ensemble learning methods and supervised learning based schemes, the highest predictive performances were achieved by proposed hybrid supervised clustering based classifier ensemble, when support vector machines were utilized as the base learners. In addition, the second highest predictive performance on all text benchmarks was obtained by proposed hybrid supervised clustering based classifier ensemble, when logistic regression classifier was utilized as the base learner.

[Table 4]

Table 4 presents the average F-measure values obtained with base learning algorithms, conventional ensemble learning methods and supervised clustering based classifier ensembles. Regarding the F-measure values presented in Table 4, supervised clustering based classifier ensembles outperform again the values obtained by conventional ensemble learning methods and the base learning algorithms. The highest F-measure values are generally achieved by proposed hybrid supervised clustering based classifier ensemble, when support vector machines were utilized as the base learners. Regarding the F-measure values, the results obtained by hybrid supervised clustering based classifier ensemble in conjunction with logistic regression and Naïve Bayes algorithms were also promising.

[Table 5]

To further evaluate results obtained in empirical analysis, we performed two-way analysis of variance (ANOVA) in the statistical program Minitab. The results of the statistical analysis for accuracy and F-measure values are presented in Table 5. According to two-way ANOVA test results, there are statistically meaningful differences between the classification accuracies and F-measure values of compared algorithms ($p < .0001$).

[Figure 3]
[Figure 4]

In Figure 3 and Figure 4, the confidence intervals for the mean values of classification accuracies and F-measure values obtained with the compared algorithms at a confidence level of 95% are presented, respectively. Figure 4 is divided into three regions, as denoted by red dashed lines, based on the statistical significances among the results. As it can be observed, the differences among the base learning algorithms and ensemble learning methods were statistically significant. In addition, the confidence interval obtained by the proposed hybrid supervised clustering based classifier ensemble (in conjunction with support vector machines) was deployed in another region of the interval plot. Hence, the higher predictive performance obtained by this classification scheme was statistically significant. Regarding the confidence intervals for F-measure values, the patterns that occurred in accuracy plots are still valid for F-measure plots. Hence, the higher predictive performance obtained by proposed hybrid supervised clustering based classifier ensemble (in conjunction with support vector machines) was also statistically significant in terms of F-measure values.

[Figure 5]

In Figure 5, average execution times of compared algorithms have been presented in seconds. As it can be seen from Figure 5, average execution times on base learning algorithms (such as Naïve Bayes, support vector machines, logistic regression and C4.5) are the lowest. Compared to the base learning algorithms, conventional ensemble learning methods and supervised clustering based ensemble schemes require more execution times. The highest execution time is involved in SAKM. Hence, there is a trade-off between the predictive performance and execution times. Although conventional ensemble learning methods and supervised clustering based ensemble schemes require more execution times, the algorithms can still be used as viable tools in classification.

6. Conclusion

This paper presents an ensemble classification approach for text classification based on hybrid supervised clustering. In the proposed ensemble classification scheme, supervised clustering is employed to obtain diversified training subsets from the original data so that the base learning algorithms with high diversity and classification accuracy can be achieved, which are the two principal factors in building robust classifier ensembles with high predictive performance. In the supervised clustering based training subset manipulation phase, a novel supervised clustering algorithm, which combines cuckoo search algorithm with supervised k-means algorithm, is introduced. The cuckoo search algorithm is utilized to assign optimal weight values for the supervised k-means algorithm. Then, each classification algorithm is trained on the diversified training subsets and the final prediction of the ensemble is obtained by combining the predictions of individual classifiers by the majority voting. The experimental analysis is conducted on eleven text classification benchmarks. The experimental results on base learning algorithms, conventional ensemble learning methods and supervised clustering based classifier ensembles indicate that ensemble learning can yield more promising results compared to the base learning algorithms and supervised clustering based classifier ensembles can further enhance the predictive performance of ensemble learning schemes. Using the 10-fold cross validation scheme, the proposed scheme achieved a classification accuracy of 97.92% for Classic4 dataset. Hence, the proposed classification scheme can be used as a viable classifier in text classification.

The main contributions of the paper can be summarized as follows: First, it presents an efficient ensemble classification scheme based on hybrid supervised clustering for text classification. The paradigm of ensemble learning for text classification has been well studied in the literature. However, the performance enhancement in ensemble text classification based on achieving high diversity remains underexplored. To fill this gap, the paper presented an extensive empirical analysis on text classification with different supervised clustering based classifier ensembles and presented a novel classifier ensemble approach which utilizes hybrid supervised clustering (based on cuckoo search and supervised k-means). The immense quantity of data available on the web is stored as text documents. Hence, efficient processing of text documents can serve as an important tool for building decision support systems and providing business intelligence. The primary focus of the proposed scheme is on text classification. However, the same technique may yield promising results on different domains with non-textual datasets.

Several limitations characterize our research. First, the presented classifier ensemble utilized the latent Dirichlet allocation model to represent text documents in an efficient way. In the experimental analysis, other topic modelling approaches or representation schemes are not taken into consideration. Secondly, the performance of simulated annealing and cuckoo search algorithms are presented in the training set manipulation phase. Though the predictive performances obtained by metaheuristic algorithms are promising, their performance may be further enhanced with the parameter tuning.

As the limitations of the study suggests, there are several directions to extend the research in the future. First, the predictive performance of the proposed classifier ensemble in conjunction with different text representation schemes can be taken into consideration. Moreover, the predictive performance of other metaheuristic approaches can be evaluated in the training set manipulation phase. By taking these issues into account, the predictive performance of proposed scheme may be further improved.

References

Aggarwal, C.C. and Zhai, C.X. (2012), "A survey of text classification algorithms", in C.C. Aggarwal and C.X. Zhai (Eds.), *Mining text data,* Springer, Berlin, pp. 77-128.

Al-Harbi, S.H. and Rayward-Smith, V.J. (2006), "Adapting k-means for supervised clustering", *Applied Intelligence,* Vol. 24*,* pp. 219-226.

Al-Salemi, B., Noah, S.A.M. and Aziz, M.J. A. (2016), "RFBoost: an improved multi-label boosting algorithm and its application to text categorization", *Knowledge-Based Systems,* Vol. 103, pp. 104-117.

Blei, D.M. (2012), "Probabilistic topic models", *Communications of the ACM,* Vol. 55, No. 4, pp. 77-84.

Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003), "Latent Dirichlet allocation", *Journal of Machine Learning Research,* Vol. 3, pp. 993-1022.

Breiman, L. (1996), "Bagging predictors", *Machine Learning,* Vol. 4 No. 2, pp. 123-140.

Cardoso, D.O., Carvalho, D.S., Alves, D.S.F., Souza, D.F.P., Carneiro, H.C.C., Pedreira, C.E., Lima, P.M.V. and França, F.M.G. (2016), "Financial credit analysis via a clustering weightless neural classifier", *Neurocomputing,* Vol. 183*,* pp. 70-78.

Cruz, D.P.F., Maia, R.D., da Silva, L.A. and de Castro, L.N. (2016), "BeeRBF: A bee-inspired data clustering approach to design RBF neural network classifiers", *Neurocomputing,* Vol. 172, pp. 427-437.

Ditterich, T.G. (2000), "Ensemble methods in machine learning", In J. Kittler et al. (Eds.), *Multiple Classifier Systems,* Springer-Verlag, Berlin, pp. 1-15.

Eick, C.F., Zeidat, N. and Zhao, Z. (2004), "Supervised clustering- algorithms and benefits", in *Proceedings of IEEE International Conference on Tools with Artificial Intelligence, Florida*, *2004,* IEEE, New York, pp. 774-776.

Elghazel, H., Aussem, A., Gharroudi, O. and Saadaoui, W. (2016), "Ensemble multi-label text categorization based on rotation forest and latent semantic indexing", *Expert Systems with Applications,* Vol. 57, pp. 1-11.

Farahbod, F. and Eftekhari, M. (2013), "A new clustering-based approach for modeling fuzzy rule-based classification systems", *Transactions of Electrical Engineering,* Vol. 37*,* pp. 67-77.

Finley, T. and Joachims, T. (2008), "Supervised k-means clustering*"*, working paper, Cornell Computing and Information Science.

Freund, Y., and Schapire, R.E. (1996), "Experiments with a new boosting algorithm", in *Proceedings of the Thirteenth International Conference*, *Bari, Italy*, *1996,* ACM, New York, pp. 325–332.

Gandomi, A.H., Yang, X.S. and Alavi, A.H. (2013), "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems", *Engineering with Computers*, Vol. 29, pp. 17-35.

Haghighi, M.S., Vahedian, A. and Yazdi, H.S. (2012), "Making diversity enhancement based on multiple classifier system by weight tuning", *Neural Processing Letters,* Vol. 35, pp. 61-80.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H. (2009), "The Weka data mining software: An update", *SIGKDD Explorations*, Vol. 11 No. 1, pp. 10-18.

Ho, T.K. (1998), "The random subspace method for constructing decision forests". *IEEE Transactions on Pattern Analysis and Machine Learning,* Vol. 22 No. 8, pp. 832–844.

Jain, A.K. (2010), "Data clustering: 50 years beyond k-means", *Pattern Recognition Letters,* Vol. 31*,* pp. 651-666.

John, G.H. and Langley, P. (1995), "Estimating continuous distributions in Bayesian classifiers", in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, *San Francisco, CA,* IEEE, New York, pp. 338–345.

Kantardzic, M. (2011), *Data mining: Concepts, models, methods and algorithms*. Wiley, New York.

Kuncheva, L. (2005), *Combining pattern classifiers: Methods and algorithms,* Wiley, New York.

Lee, S.L.A., Kouzani, A.Z. and Hu, E.J. (2010), "Random forest based lung nodule classification aided by clustering", *Computerized Medical Imaging and Graphics,* Vol. 340*,* pp. 535-542.

Li, W., Miao, D. and Wang, W. (2011), "Two-level hierarchical combination method for text classification", *Expert Systems with Applications,* Vol. 38, pp. 2030-2039.

Lin, C., Chen, W., Qiu, C., Wu, Y., Krishnan, S. and Zhou, Q. (2014), "LibD3C: ensemble classifiers with a clustering and dynamic selection strategy", *Neurocomputing,* Vol. 123*,* pp. 424-435.

Lochter, J.V., Zanetti, R.F., Reller, D. and Almeida, T.A. (2016), "Short text opinion detection using ensemble of classifiers and semantic indexing", *Expert Systems with Applications,* Vol. 62, pp. 243-249.

Mendes-Moreira, J., Soares, C., Jorge, A.M. and De Sousa, J.F. (2012), "Ensemble approaches for regression: a survey", *ACM Computing Surveys,* Vol. 45 No. 1, pp. 10-40.

Meng, J., Hao, H. and Luan, Y. (2016), "Classifier ensemble selection based on affinity propagation clustering", *Journal of Biomedical Informatics,* Vol. 60*,* pp. 234-242.

Mimno, D., Wallach, H.M., Talley, E., Leenders, M. and McCallum, A. (2011), "Optimizing semantic coherence in topic models", In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011,* ACM, Stroudsburg, pp. 262-272.

Mohamad, A.B., Zain, A.M. and Bazin, N.E.N. (2014), "Cuckoo search algorithm for optimization problems-a literature review and its applications", *Applied Artificial Intelligence,* Vol. 28, pp. 419-448.

Onan, A. (2016), "An ensemble scheme based on language function analysis and feature engineering for text genre classification", *Journal of Information Science.*

Onan, A., Korukoğlu, S. and Bulut, H. (2016b), "Ensemble of keyword extraction methods and classifiers in text classification", *Expert Systems with Applications,* Vol. 57, pp. 232-247.

Onan, A., Korukoğlu, S. and Bulut, H. (2016c), "A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification", *Expert Systems with Applications,* Vol. 62, pp. 1-16.

Perikos, I. and Hatzilygeroudis, I. (2016), "Recognizing emotions in text using ensemble of classifiers", *Engineering Applications of Artificial Intelligence,* Vol. 51, pp. 191-201.

Quinlan, R. (1993), *C4.5: Programs for Machine learning,* Morgan Kaufmann, San Mateo.
Rahman, A. and Verma, B. (2013), "Ensemble classifier generation using non-uniform layered clustering and genetic algorithm", *Knowledge-Based Systems,* Vol. 43*,* pp. 30-42.

Rodger, J.A. (2015), "Discovery of medical big data analytics: improving the prediction of traumatic brain injury survival rates by datamining patient information processing software hybrid hadoop hive", *Informatics in Medicine Unlocked*, Vol. 1, pp. 17-26.

Rossi, R.G., Maraccini, R.M. and Rezende, S.O. (2013), *"Benchmarking text collections for classification and clustering tasks",* working paper, University of Sao Paulo.

Sebastiani, F. (2002), "Machine learning in automated text categorization", *ACM Computing Surveys,* Vol. 34 No. 1*,* pp. 1-47.

Sebastiani, F. (2005), "Text categorization", in A. Zanasi (Ed.), *Text Mining and its Applications,* WIT Press, Southampton, pp. 109-129.

Shi, L., Ma, X., Xi, L., Duan, Q. and Zhao, J. (2011), "Rough set and ensemble learning based semi-supervised algorithm for text classification", *Expert Systems with Applications,* Vol. 38, pp. 6300-6306.

Song, G., Ye, Y., Zhang, H., Xu, X., Lau, R.Y.K. and Liu, F. (2016), "Dynamic clustering forest: an ensemble framework to efficiently classify textual data stream with concept drift", *Information Sciences,* Vol. 357*,* pp. 125-143.

Tsai, C-F. (2014), "Combining cluster analysis with classifier ensembles to predict financial distress", *Information Fusion,* Vol. 16*,* pp. 46-58.

Vapnik, V. (1995), *The nature of statistical learning theory,* Springer, New York, NY.

Vucetic, S. and Obradovic, Z. (2000), "Discovering homogeneous regions in spatial data through competition", in *Proceedings of the 17th International Conference on Machine Learning, USA, 2000,* ACM, New York, pp. 1091-1098.

Wang, F., He, X-S., Wang, Y. and Yang, S.M. (2012), "Markov model and convergence analysis based on cuckoo search algorithm", *Computer Engineering*, Vol. 38, pp. 180-185.

Wang, G., Sun, J., Ma, J., Xu, K. and Gu, J. (2014), "Sentiment classification: the contribution of ensemble learning", *Decision Support Systems,* Vol. 57, pp. 77-93.

Wang, J., Zhou, B. and Zhou, S. (2016), "An improved cuckoo search optimization algorithm for the problem of chaotic systems parameter estimation", *Computational Intelligence and Neuroscience,* Vol. 2016*,* pp. 1-8.

Xia, R., Zong, C., and Li, S. (2011), "Ensemble of feature sets and classification algorithms for sentiment classification", *Information Sciences,* Vol. 181, pp. 1138-1152.

Xiao, H., Xiao, Z. and Wang, X.Y. (2016), "Ensemble classification based on supervised clustering for credit scoring", *Applied Soft Computing,* Vol. 43*,* pp. 73-86.

Yang, B., Zhang, Y. and Li, X. (2011), "Classifying text streams by keywords using classifier ensemble", *Data and Knowledge Engineering,* Vol. 70*,* pp. 775-793.

Yang, B., Zhang, Y., and Li, X. (2011), "Classifying text streams by keywords using classifier ensemble", *Data & Knowledge Engineering,* Vol. 70, pp. 775-793.

Yang, X.S. and Deb, S. (2010), "Engineering optimization by cuckoo search", *International Journal of Mathematical Modelling and Numerical Optimization,* Vol. 1 No. 4, pp. 330-343.

Yang, X-S. and Deb, S. (2014), "Cuckoo search: recent advances and applications", Neural Computing and Application", *Neural Computing and Applications*, Vol. 24, pp. 169-174.

Yang, X-S. and Deb, Y.S. (2009), "Cuckoo search via Levy flights", In *Proceedings of IEEE World Congress on Nature and Biology Inspired Computing, India, 2009, IEEE,* New York, pp. 210-214.

Zhang, H. and Cao, L. (2014), "A spectral clustering based ensemble pruning approach", *Neurocomputing,* Vol. 139, pp. 289-297.

Zhang, H. and Lu, J. (2010), "Creating ensembles of classifiers via fuzzy clustering and deflection", *Fuzzy Sets and Systems,* Vol. 161 No. 13*,* pp. 1790-1802.

Zheng, N. and Xue, J. (2009), *Statistical Learning and Pattern Analysis for Image and Video Processing,* Springer, Berlin.

Zhou, Z-H. (2012), *Ensemble methods: foundations and algorithms,* Chapman and Hall, New York, NY.

```
begin
    Objective function f(x), x=(x_1, ...,x_d)^T
    Generate initial population of n host nests x_i (i=1,2,…,n)
    while (t<MaxGeneration) or (stopping criterion)
        Get a cuckoo randomly by Levy flights
            Evaluate its quality/fitness F_i
        Choose a nest (j) among n randomly
        If (F_i> F_j)
            Replace j by the new solution
        End
        A fraction (p_a) of worse nests are abandoned and new ones are built
        Keep the best solutions (or nests with quality solutions)
        Rank the solutions and find the current best
    end while
end
```

Figure 1. The general structure of cuckoo search algorithm (Yang and Deb, 2009).

Training subset manipulation phase

1. Initially assign a random weight for each instance of the training set.
2. Run supervised k-means algorithm with the weighted Euclidean metric.
3. Compute the fitness of the partitions using Equation 4.
4. Adjust the weight values of each instance by cuckoo search algorithm.
5. Repeat Step 3 and Step 4 until stopping criterion has been reached.
6. Obtain clusters by the supervised clustering corresponding to instances with different class labels.
7. Divide each cluster into *N* subsets.
8. Obtain diversified training subsets by pair-wisely combining one subset from each cluster.

Training phase

1. Train each base learning algorithm on diversified training subsets (using different subsets for each base learning algorithm).

Ensemble combination phase

1. Combine the predictions of base learning algorithms by majority voting.

Figure 2. The proposed ensemble classification scheme.

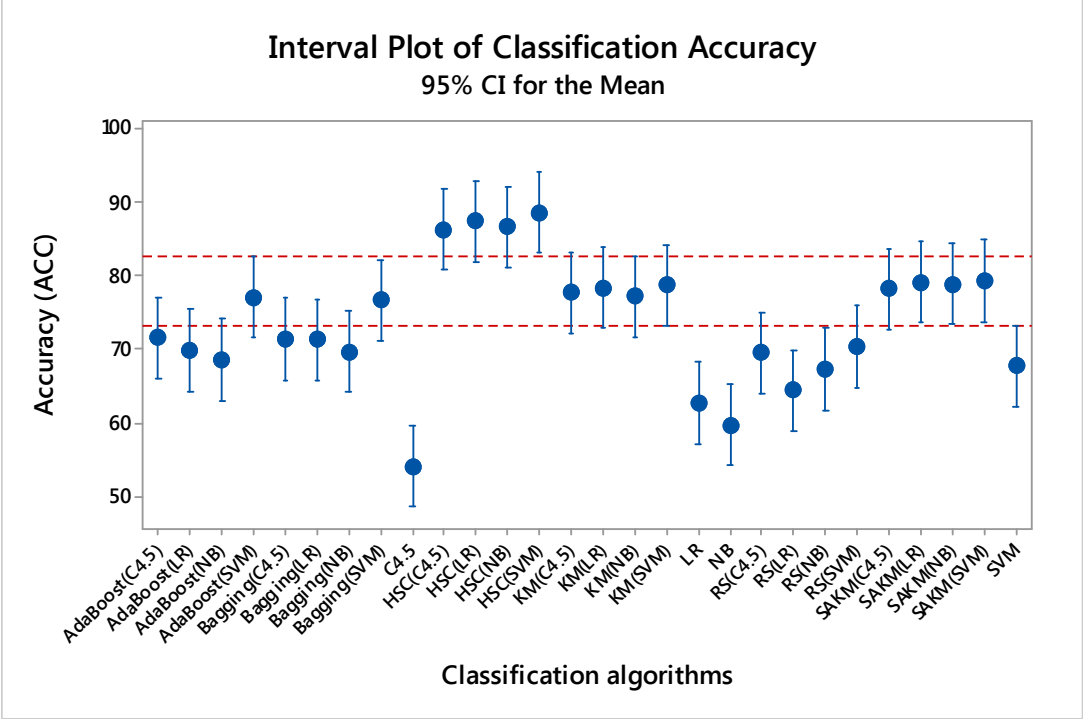

Figure 3. Interval plots of classification accuracies for the compared algorithms.

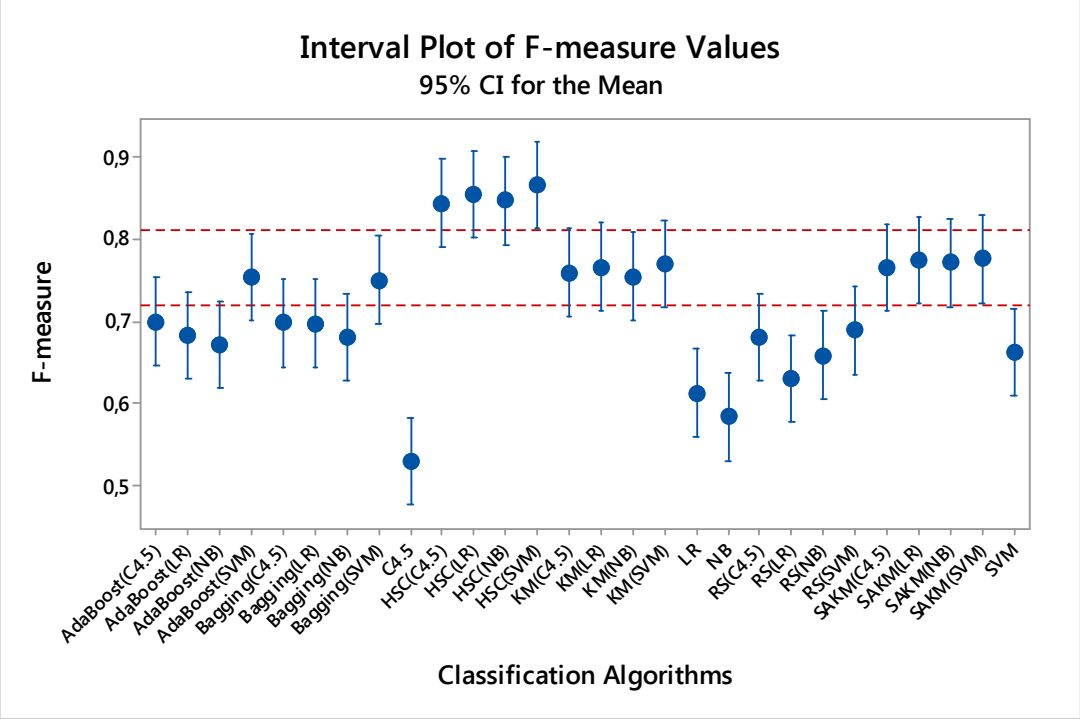

Figure 4. Interval plots of F-measure values for the compared algorithms.

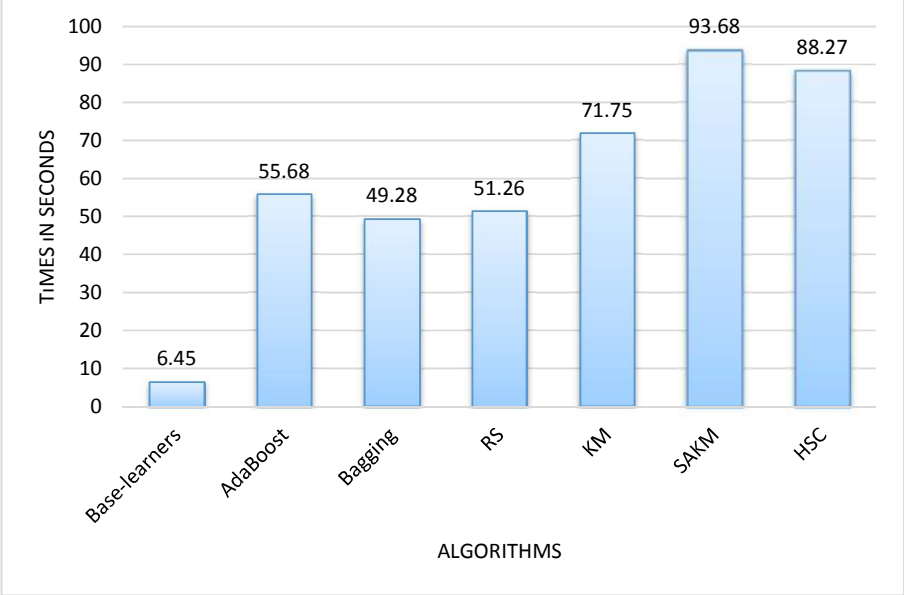

Figure 5. Execution times of algorithms in seconds.

Table 1. Descriptive information for text collections (Rossi et al., 2013)

Dataset	Domain	Number of documents	Number of features	Number of classes
20-ng	E-mails	18808	45434	20
ACM	Scientific	3493	60768	40
Classic4	Abstracts	7095	7749	4
CSTR	Scientific	299	1726	4
DMOZ-Business-500	Web pages	18500	8303	37
DMOZ-Computers-500	Web pages	9500	5011	19
DMOZ-Health-500	Web pages	6500	4217	13
DMOZ-Science-500	Web pages	6000	4821	12
DMOZ-Sports-500	Web pages	13500	5682	27
Enron	E-mails	13199	18194	20
Hi-tech	News Articles	2301	12942	6

Table 2. Parameter values for the metaheuristic schemes.

Algorithm	Parameters
Cuckoo search algorithm	Chaotic coefficient=4.0, Chaotic type: Logistic map, Number of iterations=100, Mutation type: bit-flip, Objective type: Merits, p_a=0.25, population size=20, seed=1, sigma= 0.69657
Simulated annealing algorithm	Maximum value of temperature (T_{max}) = 100, Minimum value of temperature (T_{min}) = 0.00001, Soft limit (SL) = 200, Hard limit (HL) = 100, cooling rate (α) = 0.8, Number of iterations = 100

Table 3. Classification accuracies obtained with classifiers and ensemble methods.

Algorithm	20-ng	ACM	CSTR	DM1	DM2	DM3	DM4	DM5	ENR	HiT	CL4
NB	67.85	55.26	73.80	39.24	47.06	62.62	50.42	52.61	53.38	60.25	92.52
LR	76.31	68.88	42.73	47.06	54.25	67.67	56.37	64.63	59.16	64.13	86.88
SVM	77.22	70.89	70.04	48.07	55.04	69.31	57.01	66.08	64.24	70.11	95.14
C4.5	64.72	47.79	61.81	32.97	39.73	57.90	40.30	54.66	54.98	52.19	86.51
AdaBoost(NB)	71.85	58.94	74.73	55.24	61.06	76.62	64.42	64.61	63.38	71.25	91.02
AdaBoost(LR)	80.14	70.71	49.31	60.28	60.35	75.74	60.45	72.63	69.23	75.35	92.74
AdaBoost(SVM)	80.90	75.29	80.05	63.73	68.46	82.78	70.65	77.80	74.28	79.52	93.40
AdaBoost(C4.5)	77.39	66.95	71.48	56.49	60.95	78.64	63.74	73.51	71.15	73.31	92.26
Bagging(NB)	72.36	61.91	78.00	55.40	61.49	77.25	64.74	64.87	64.48	72.97	91.08
Bagging(LR)	80.26	72.56	47.22	62.44	66.61	80.66	68.31	75.91	69.05	75.08	85.14
Bagging(SVM)	81.21	74.68	72.36	64.05	68.95	83.26	71.06	78.06	74.28	81.03	93.53
Bagging(C4.5)	76.61	64.26	72.25	58.17	62.12	78.36	63.90	73.81	70.96	73.21	89.97
RS(NB)	69.14	59.60	77.40	53.67	59.50	73.67	62.38	63.69	60.02	69.40	90.66
RS(LR)	75.47	64.89	46.54	56.36	62.43	75.91	65.09	68.29	61.14	67.21	63.79
RS(SVM)	75.34	68.18	61.67	57.29	63.18	77.06	65.74	69.24	68.20	78.57	88.48
RS(C4.5)	73.70	64.75	72.86	54.79	58.87	75.16	60.67	71.17	68.32	72.58	90.56
KM(NB)	82.12	75.86	75.87	63.19	70.28	78.71	70.76	80.06	77.94	81.60	91.32
KM(LR)	84.63	77.20	76.74	65.91	70.54	79.37	71.62	80.49	78.54	83.46	91.94
KM(SVM)	84.97	77.36	77.38	66.31	70.79	79.78	71.96	80.80	79.40	83.75	92.15
KM(C4.5)	81.46	75.86	75.93	65.28	70.50	78.86	70.81	80.16	78.46	83.19	92.40
SAKM(NB)	84.22	77.44	77.61	67.01	70.92	79.92	72.08	80.98	79.72	83.82	92.32
SAKM(LR)	85.67	77.79	78.02	67.12	71.14	80.33	72.13	81.15	79.84	83.87	92.42
SAKM(SVM)	85.68	78.25	78.06	67.14	71.45	80.55	72.17	81.23	79.90	84.14	92.52
SAKM(C4.5)	81.56	77.37	76.88	65.95	70.76	79.53	71.82	80.55	78.76	83.49	92.01
HSC(NB)	94.23	87.33	85.18	74.98	78.81	88.05	79.93	88.29	87.05	91.90	95.59
HSC(LR)	***95.06***	***91.34***	***86.27***	***75.63***	***78.93***	***88.05***	***80.17***	***88.40***	***87.08***	***92.64***	***96.73***
HSC(SVM)	**96.41**	**92.83**	**86.94**	**76.41**	**79.15**	**89.64**	**82.37**	**89.78**	**88.27**	**93.72**	**97.92**
HSC(C4.5)	92.96	86.26	85.15	74.77	78.55	87.96	79.29	88.25	86.74	91.46	96.58

DM1: DMOZ-Business-500 dataset, DM2: DMOZ-Computers-500 dataset, DM3: DMOZ-Health-500 dataset, DM4: DMOZ-Science-500 dataset, DM5: DMOZ-Sports-500 dataset, ENR: Enron dataset, HiT: Hi-tech dataset, CL4: Classic4 dataset, NB: Naïve Bayes, LR: Logistic regression, SVM: Support vector machines, RS: Random Subspace, KM: Supervised k-means based classifier ensemble, SAKM: simulated annealing and supervised k-means based classifier ensemble, HSC: proposed hybrid supervised clustering based classifier ensemble.

Table 4. F_1-measure values obtained with classifiers and ensemble methods.

Algorithm	20-ng	ACM	CSTR	DM1	DM2	DM3	DM4	DM5	ENR	HiT	CL4
NB	0.64	0.55	0.72	0.38	0.47	0.63	0.49	0.52	0.52	0.59	0.91
LR	0.72	0.68	0.42	0.46	0.54	0.68	0.55	0.63	0.58	0.63	0.85
SVM	0.73	0.70	0.69	0.47	0.54	0.69	0.56	0.65	0.63	0.69	0.93
C4.5	0.61	0.47	0.61	0.32	0.39	0.58	0.39	0.54	0.54	0.51	0.85
AdaBoost(NB)	0.68	0.58	0.73	0.54	0.60	0.77	0.63	0.63	0.62	0.70	0.89
AdaBoost(LR)	0.75	0.70	0.48	0.59	0.60	0.76	0.59	0.71	0.68	0.74	0.91
AdaBoost(SVM)	0.76	0.75	0.78	0.62	0.68	0.83	0.69	0.76	0.73	0.78	0.92
AdaBoost(C4.5)	0.73	0.66	0.70	0.55	0.60	0.79	0.62	0.72	0.70	0.72	0.90
Bagging(NB)	0.68	0.61	0.76	0.54	0.61	0.77	0.63	0.64	0.63	0.72	0.89
Bagging(LR)	0.76	0.72	0.46	0.61	0.66	0.81	0.67	0.74	0.68	0.74	0.83
Bagging(SVM)	0.76	0.74	0.71	0.63	0.68	0.83	0.70	0.76	0.73	0.79	0.92
Bagging(C4.5)	0.72	0.64	0.71	0.57	0.61	0.78	0.63	0.72	0.70	0.72	0.88
RS (NB)	0.65	0.59	0.76	0.53	0.59	0.74	0.61	0.62	0.59	0.68	0.89
RS (LR)	0.71	0.64	0.46	0.55	0.62	0.76	0.64	0.67	0.60	0.66	0.63
RS (SVM)	0.71	0.67	0.60	0.56	0.63	0.77	0.64	0.68	0.67	0.77	0.87
RS (C4.5)	0.69	0.64	0.71	0.54	0.58	0.75	0.59	0.70	0.67	0.71	0.89
KM(NB)	0.77	0.75	0.74	0.62	0.70	0.79	0.69	0.78	0.76	0.80	0.89
KM(LR)	0.80	0.76	0.75	0.65	0.70	0.79	0.70	0.79	0.77	0.82	0.90
KM(SVM)	0.80	0.77	0.76	0.65	0.70	0.80	0.71	0.79	0.78	0.82	0.90
KM(C4.5)	0.77	0.75	0.74	0.64	0.70	0.79	0.69	0.79	0.77	0.82	0.91
SAKM(NB)	0.79	0.77	0.76	0.66	0.70	0.80	0.71	0.79	0.78	0.82	0.90
SAKM(LR)	0.81	0.77	0.76	0.66	0.70	0.80	0.71	0.80	0.78	0.82	0.91
SAKM(SVM)	0.81	0.77	0.76	0.66	***0.71***	0.81	0.71	0.80	0.78	0.82	0.91
SAKM(C4.5)	0.77	0.77	0.75	0.65	0.70	0.79	0.70	0.79	0.77	0.82	0.90
HSC(NB)	***0.89***	0.86	0.83	0.73	**0.78**	***0.88***	0.78	***0.87***	***0.85***	0.90	0.94
HSC(LR)	***0.89***	***0.90***	**0.85**	***0.74***	**0.78**	***0.88***	***0.79***	***0.87***	***0.85***	***0.91***	***0.95***
HSC(SVM)	**0.91**	**0.92**	**0.85**	**0.75**	**0.78**	**0.90**	**0.81**	**0.88**	**0.87**	**0.92**	**0.96**
HSC(C4.5)	0.87	0.85	0.83	0.73	**0.78**	***0.88***	0.78	0.86	***0.85***	0.90	***0.95***

Table 5. Two-way analysis of variance results.

Accuracy Values					
Source	**DF**	**SS**	**MS**	**F**	**p**
Algorithms	27	20978	776.95	37.69	0.000
Datasets	10	18745	1874.51	90.94	0.000
Error	270	5565	20.61		
Total	307	45288			
F_1-measure Values					
Source	**DF**	**SS**	**MS**	**F**	**p**
Algorithms	27	2.0157	0.074657	37.59	0.000
Datasets	10	1.7238	0.172383	86.81	0.000
Error	270	0.5362	0.001986		
Total	307	4.2757			

DF: degrees of freedom, SS: adjusted sum of squares, MS: adjusted mean square, F: F-value, p: probability value.