The Effectiveness of Homogenous Ensemble Classifiers for Turkish and English Texts

Zeynep Hilal Kilimci[1], Selim Akyokus[2], Sevinc Ilhan Omurca[3]
Computer Engineering Department[1, 2, 3]
{Dogus University[1, 2], Kocaeli University[3]}
{Istanbul, 34722, Turkey[1, 2], Umuttepe Yerleskesi, Kocaeli, 41380, Turkey[3]}
{hkilimci, sakyokus@dogus.edu.tr[1,2], silhan@kocaeli.edu.tr[3]}

***Abstract*—Text categorization has become more and more popular and important problem day by day because of the large proliferation of documents in many fields. To come up with this problem, several machine learning techniques are used for categorization such as naïve Bayes, support vector machines, artificial neural networks, etc. In this study, we concentrate on ensemble of multiple classifiers instead of using only a single one. We perform a comparative analysis of the impact of the ensemble techniques for text categorization domain. To carry out this, the same type of base classifiers but diversified training sets are used which is referred as homogenous ensembles. In order to diversify the training dataset, various ensemble algorithms are utilized such as Bagging, Boosting, Random Subspace and Random Forest. Multivariate Bernoulli Naïve Bayes is preferred as a base classifier due to its superior classification performance compared to the success of the other single classifiers. A wide range of comparative and extensive empirical studies are conducted on four widely-used datasets in text categorization domain in both Turkish and English. Finally, the effectiveness of ensemble algorithms is discussed.**

Keywords—Ensemble learning; homogeneous ensembles; bagging; random subspace; random forest; text categorization.

I. INTRODUCTION

Text categorization/classification has always been an important research topic because of very large amount of text documents available in different types of application areas today. The objective of text categorization is to classify a given document into one of predefined categories by the use of machine learning techniques. For text categorization, the supervised learning techniques are usually used to build classifiers from a set of training documents. From the training set, a classifier learns and forms a model of relationship between features and class labels (categories). Once trained, the classifier can be used to determine the category of a new document from a test dataset.

The text categorization process usually involves parsing of documents, tokenization, stop-words removal, stemming, and representation of documents in appropriate formats and weights, feature reduction, selection of classifiers (learning algorithms), training and testing. To represent the documents, the bag of words model is commonly used in document categorization. In bag of words model, the document set is represented as a document-term matrix where each row describes a document and each column corresponds to a term (word). Each entry in the matrix contains a weight that reflects the importance of a term with respect to document collection. Different term weighting approaches such as term frequency and TF-IDF can be used to represent each weight. An excellent review of text categorization algorithms is given in papers [1, 2]. The classification algorithms such as naïve Bayes (NB), k-nearest neighbors (k-NN), decision trees (DT), artificial neural networks (ANN) and support vector machines (SVM) are commonly used for text classification.

In machine learning, ensemble methods use multiple learning algorithms (classifiers) to achieve a better performance than a single classifier. The systems that use ensemble methods are also called multiple classifier systems, ensemble-based classifiers or just ensemble systems. An ensemble-based classifier is a group of combined individual classifiers. Each individual classifier is called a base or weak classifier (learner). During training, each base classifier is trained separately on a given training data set. An ensemble approach is usually composed of an ensemble generation and integration (aggregation, combination or fusion) steps. In ensemble generation step, a diverse set of base classifiers is generated from the training data set. In the integration step, the outputs of the trained base classifiers are integrated to obtain a final decision. The main strategy in ensemble approach is therefore to generate many classifiers and integrate outputs of classifier such that the combination of classifiers improves the performance of a single classifier [3-6].

The success of an ensemble system depends on the diversity of the base classifiers that make up the ensemble. Each base classifier must exhibit some level of diversity among themselves. The diversity can be achieved by mainly three approaches: data diversity, parameter diversity and structural diversity [6]. In data diversity, different training data subsets are generated from the original dataset for each of base classifier through re-sampling techniques. Parameter diversity approach uses different training parameter for different classifiers. For example, a neural network can be trained with different layers, initial weights and learning rates. The structural diversity is achieved by using different learning algorithms. If all base classifiers are generated by using the same learning algorithm, the ensemble system is called homogeneous, otherwise it is called heterogeneous. Heterogeneous ensemble systems use more than one learning algorithm to realize diversity.

978-1-4673-9910-4/16/$31.00 ©2016 IEEE

Compared with other classification problems, text categorization problem has several distinct characteristics such as the high dimensionality of the input space, sparsity of document vectors, and scarceness of irrelevant features [7]. There has been a limited research on the use of ensemble systems on text categorization. In this paper, we attempt to investigate the classification success of homogenous ensemble classifiers by providing data diversity with various ensemble algorithms on English and Turkish text documents. An extensive experiment results demonstrate that homogenous ensemble learners boost the classification performance effectively compared to the single classifiers on text categorization field.

The rest of the paper is organized as follows: Section 2 gives related researches on the use of ensemble systems on text categorization. Section 3 presents base learners and ensemble techniques used in experimental studies. Experimental setup and results are given in sections 4 and 5. Section 6 concludes the paper with a discussion.

II. Related Work

In this section, we try to review some of research that uses ensemble techniques for text categorization. In [8], authors use a naïve Bayes classifier called moderated asymmetric naïve Bayes classifiers (MANB) as a base classifier in their homogeneous ensemble setting. They use ensemble methods called k-fold partitioning, bagging, boosting, biased k-partitioning, biased k-fold partitioning and biased clustering on a single data set. They also test a heterogeneous ensemble method by combining NB and SVM learning algorithms.

A pairwise ensemble approach is presented in [9] that achieve better performance than popular ensemble approaches bagging and ECOC. An ensemble classifier for Twitter sentiment analysis is proposed in [10] where the dataset includes very short texts. A combination of several polarity classifiers provides an improvement of the base classifiers. In [11], authors combined SVM, k-NN and Rocchio classifiers by using Dempster's rule of combination and observed a better performance than base classifiers. In [12], Random subspace method is applied to text categorization problem. The paper focuses the estimation of ensemble parameters of size and the dimensionality of each random subspace submitted to the base classifiers.

Authors in [13] applied ensemble methods to multi-class text documents where each document can belong to more than on category. They evaluated the performance of ensemble techniques by using multi-label learning algorithms. H. Elghazel et al. [14] resents a novel ensemble multi-label text categorization algorithm, called Multi Label Rotation Forest (MLRF) that uses a combination of Rotation Forest and Latent Semantic Indexing.

In a recent study [15], Onan et al empirically evaluate the predictive performance of ensemble learning methods on text documents that are represented keywords. They first apply different keyword extraction algorithms to test dataset. Then, they evaluate the performance of five different ensemble methods that use four different base classifiers on the documents represented by keywords.

III. Base Learners and Ensemble Techniques

This section briefly reviews learning algorithms and ensemble techniques used in experiments. Bernoulli Naive Bayes learning algorithm is selected as a base learner in our study after comparing several classifiers MVNB (multivariate Bernoulli naïve Bayes), MNB (multinomial naïve Bayes), SVM (support vector machines), and DT (decision trees) on our datasets. We applied bagging, boosting, random subspace, and random forest ensemble techniques.

A Naïve Bayes classifier is a simple probabilistic classifier that uses Bayes' theorem with independence assumption of features. Two frequently models used for text categorization are called multivariate Bernoulli naïve Bayes (MVNB) and multinomial naïve Bayes (MNB). MNB represents each document by the frequency of words that appears in the document. In MVNB, each document is represented by a feature vector with binary variables that can takes values 1 or 0 depending upon the presence of a word in the document. MVNB classifier is expressed in In Eq. (1) where occurrence of term t in document i is indicated by B_{it} which can be either 1 or 0. $|D|$ indicates the number of labeled training documents. Formula is given using Laplace smoothing. $P(c_j|d_i)$ is 1 if document i is in class j. Finally, the probability of term w_t in class c_j is [16]:

$$P(w_t \mid c_j) = \frac{1 + \sum_{i=1}^{|D|} B_{it} P(c_j \mid d_i)}{2 + \sum_{i=1}^{|D|} P(c_j \mid d_i)} \quad (1)$$

Support vector machines (SVM) are one of the best supervised machine learning algorithms based on the concept of decision planes that separate set of instances of two different classes [17]. It is a linear classifier, but can be used for a non-linear classification using a method called the kernel trick that implicitly maps input instances into high-dimensional feature spaces. It is an effective algorithm in high dimensional spaces. SVM can use different kernel functions that enable us to obtain a set of diverse classifiers with different decision boundaries. Decision trees (DTs) are a supervised learning method used for classification and regression [18]. Decision trees build classification or regression models in the form of a tree structure. A decision tree is built in top-down manner that breaks down a dataset into smaller and smaller subsets by using splitting criteria. It is a simple algorithm that enables easy interpretation and visualization of relationships among dataset features.

Bagging is one of the most popular and earliest ensemble based algorithms [19]. It is an abbreviation for bootstrap aggregating. Diversity is achieved by means of resampling in which different training data subsets are randomly drawn with replacement from the entire training dataset. Each data subset is used to train a different learner in the set of ensemble learners. It reaches a final decision by combining decisions of individual weak learners by taking a majority vote.

Boosting algorithm is considered as one of the most significant progresses in the recent history of machine learning

[20]. The main idea in boosting is to generate a set of learners that uses a data subset in which each instance is associated with a weight. It works by repeatedly running weak learners on various distributions over the training data. At the beginning, all instances have an equal weight. At each iteration depending upon the training error of previous classifiers, the weights of misclassified instances are updated. Each classifier uses a subset of instances drawn from an updated distribution of training dataset. At each step, instances that are incorrectly predicted by previous classifiers are chosen more often than instances that were correctly predicted. The final decision is obtained by weighted majority voting of the classes predicted by the individual classifier. There are many variants of the boosting algorithm such as AdaBoost, AdaBoost.M1, AdaBoost.M2, AdaBoost.R, Arcing and Real Adaboost [3-6]. We used AdaBoost.M1 in our experiments.

Random subspace (RS) ensemble is similar to bagging but it selects a random subset of features from the dataset instead of instances [21, 22]. Given a dataset with d features (dimensions), RS randomly select d^* features where $d^* < d$. Selection is repeated S times to get S different feature subsets in order to cover a large portion of features in the original dataset. Then S base classifiers are trained with S feature subsets. The final decision is obtained by combining decisions of S base classifiers by a voting scheme. The RS is expected to perform well when the number of features is much larger than the number of training objects.

Random forests, introduced by Breiman [23], are a collection of decision tree classifiers. It is a particular implementation of bagging in which each base classifier is a decision tree. Bagging is used to select training subsets for each individual decision tree. The splitting criterion used in Random forests differs from standard decision trees where each node is split by the best feature among all other features. In Random forests, a random subset of features is selected first and then the best split is decided on the random subset of features. This strategy works well and provides additional randomness to the algorithm in addition to bagging. Random forests is robust to overfitting because of randomness applied in both sample and feature spaces.

IV. Experiment Setup

We use four well-accepted benchmark datasets with different sizes and properties to explore the classification performances of the ensemble algorithms. First one is called 20News-18828[1] and it has less number of documents than the original dataset because duplicate postings are removed. Furthermore, each posting headers are eliminated except from and subject headers. It contains approximately a total of 20,000 documents and is divided into twenty different categories. Due to its extensive content, majority of state-of the-art studies [24, 25, 26, 27, 28, 29, 30, 32, 33, 34, 35] also utilize 20 Newsgroups for text categorization field.

Second dataset is the WebKB[2] which covers web pages gathered from computer science departments of different universities. These web pages are constituted of seven categories (student, faculty, staff, course, project, department and other) and nearly 8,300 pages. Another version of WebKB is called WebKB4 where the number of categories is diminished up to four [24, 25, 31, 35, 36]. We used WebKB4 in our experiments.

The remaining datasets are composed of Turkish news documents that contain traditional news articles from Hurriyet and Anadolu Agency. The documents from Hurriyet[3] contain miscellaneous news between 2010 and 2011 years and includes six categories (dunya/world, ekonomi/economy, guncel/current, spor/sports, siyaset/politics, yasam/life) and there are 1,000 documents per each category.

The last dataset is Aahaber[4], used in [40], consists of concise articles broadcasted by Turkish National News Agency: Anadolu Agency. It involves news from eight categories (Turkey, world, politics, economics, sports, education, science, culture art, and environment health) and is composed of 20,000 documents from 8 categories. Characteristics of the datasets, when no preprocessing is applied, are given in Table I covering the number of classes (|C|), the number of documents (|D|) and the vocabulary size (|V|).

We pay attention to frequent terms whose document frequency is greater than two and do not carry out any stop word filtering or stemming to prevent from any bias that can be introduced by stop list or stemming algorithms.

TABLE I. Characteristics of the Datasets with no Preprocessing

Dataset	|C|	|D|	|V|
20News-18828	20	18,828	50,570
WebKB4	4	4,199	16,116
Hurriyet	6	6,000	18,280
Aahaber	8	20,000	14,395

We perform experiments by modifying the training set size that use 90%, 70%, 50%, 30%, 10%, 5%, and 1% percentages of the data for training and the rest for testing. These percentages are represented with "ts" abbreviation in order to avoid confusion with the accuracy percentages. We also run experiments using 80% training and 20% test data with no document frequency filtering. The widely known 10-fold cross validation with 5x2 approach is applied on each dataset. This approach is similar to the previous works [24, 26, 35, 37, 38, 39] where they use 80% of data for training data and 20% for test. The number of base classifiers is generally arranged between 50 and 100 [37, 38]. We preferred to set it to 100 owing to its influence on empirical results. Finally, the accuracy results of the experiments are utilized as the evaluation criteria in order to provide comparison with state of the art results.

[1] http://people.csail.mit.edu/people/jrennie/20Newsgroups
[2] http://www.cs.cmu.edu/~textlearning
[3] www.hurriyet.com.tr
[4] www.aa.com.tr

V. Experiment Results

At first, we analyze several kinds of machine learning techniques to observe the classification performances of single classifiers in order to determine the best base classifier to be used in ensemble learning.

TABLE I. The Classification Accuracies of the Single Classifiers.

Dataset	MVNB	MNB	SVM	DT
20News-18828	*88.52±0.42*	86.26±0.36	84.18±0.79	79.82±0.72
WebKB4	84.10±1.12	85.64±1.17	*89.85±0.96*	77.45±1.07
Hurriyet	*81.16±0.96*	79.78±0.78	76.58±1.31	76.92±0.56
Aahaber	*82.70±1.07*	*82.80±0.93*	79.62±1.12	79.13±1.43
Avg	**84.12±0.89**	***83.62±0.81***	***82.56±1.05***	***78.33±0.95***

Table I demonstrates the evaluation results of the single classifiers mentioned in section III: two versions of Naïve Bayes classifier, Support Vector Machines, and Decision Tree. It is significant to notice that the classification results of single classifiers are represented for training set percentage eighty in Table I. It seems clearly that MVNB is more competitive and surpasses other classification techniques. On account of this, MVNB is preferred as a base learner for implementing ensemble classifiers. We can summary the classification success order of the single classifiers like this: MVNB>MNB>SVM>DT. To observe the success of base classifiers in different training set percentages, 20News-18828 dataset is analyzed in Figure I. While accuracy values between MVNB and others are usually much closer at the smaller training set levels, the success of MVNB is more prominent at higher training set percentages especially starting from ts30 to ts90. Thus, MVNB is the preferred base classifier to be selected among other classifiers in order to construct ensemble algorithms.

After determining a base learner, various ensemble algorithms (bagging, boosting, random subspace, random forest) are utilized to diversify the training datasets. Table II shows the classification accuracy results of the four homogenous ensemble methods that use Multivariate Bernoulli Naïve Bayes model as a base classifier for a training size of 80% (ts80).

As an ensemble algorithm, bagging is not efficient to improve the classification performance for 20News-18828 and WebKB4 datasets.

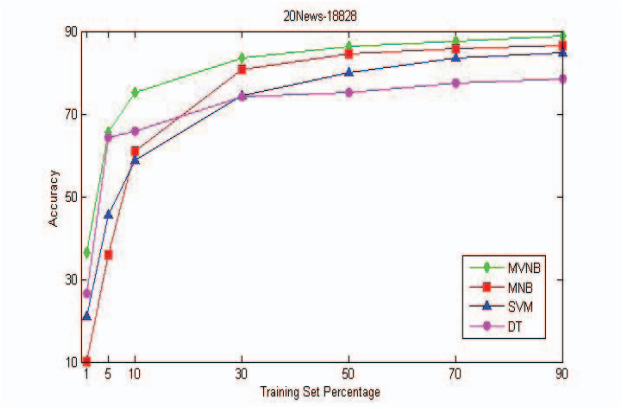


Fig. 1. Performances of the single classifiers on 20News-18828 dataset.

Bagging method decreases the accuracy results approximately 1% for 20News_18828 and slightly advances the success of MVNB classifier for WebKB4 at ts80. We do not get a significant performance increase for Turkish datasets Hurriyet and AaHaber since it provides nearly 1% classification rise for both datasets.

Random subspace demonstrates similar performance with bagging for improving classification successes. Either it does not change efficiently accuracies or it reaches maximum 2% success and increment of the classification performance is maintained via the usage of the Turkish datasets again. Thus, it can be asserted that the choice of bagging and random subspace algorithms is not significant in most cases. In order to increase the classification performance, they may be preferred for especially including agglutinative languages such as Turkish.

TABLE II. The Classification Accuracies of the Ensemble Algorithms.

Dataset	BG	BS	RS	RF
20News-18828	87.63±0.97	90.27±0.84	88.63±0.77	92.02±0.73
WebKB4	84.75±0.86	86.24±0.95	85.32±0.83	88.75±1.19
Hurriyet	82.24±1.02	81.77±0.79	83.62±0.85	84.13±0.96
Aahaber	83.16±0.88	85.91±1.05	84.69±0.96	87.35±1.37
Avg	***84.45±0.93***	***86.05±0.91***	***85.57±0.85***	***88.06±1.06***

On the other hand, boosting and random forests demonstrate obvious classification performance compared to the bagging and random subspace algorithms. While boosting provides nearly 2-3% improvement compared to the performance of the single classifier, classification success of it ranges from 1% to 3% in proportion to bagging and random subspace. On Hurriyet dataset, boosting provides a very slight improvement. In general, we can give precedence to the boosting algorithm instead of bagging and random subspace when determining ensemble techniques for all datasets.

A close inspection of Table II reveals that random forest algorithm is better than the other ensemble algorithms in terms the classification success and it overwhelmingly outperforms

others for all datasets at ts80. Random forest provides the significant contribution to the classification performance and it demonstrates minimum 3% and maximum 5% rise compared to the success of the single classifier, bagging, and random subspace algorithms. When observing the average values of the accuracies, its superior performance is seen explicitly and the classification success of them can be ordered as: RF>BS>RS>BG. As a result of this, random forest should be given the first preference among other ensemble algorithms in order to improve classification performance on text categorization domain.

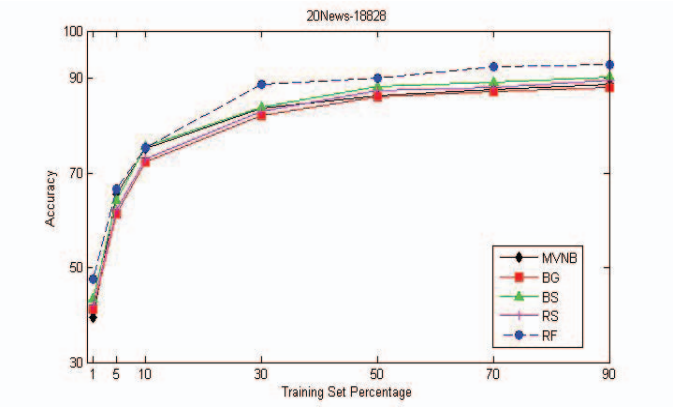


Fig. 2. Performance comparison of the ensemble classifiers on 20News-18828 dataset.

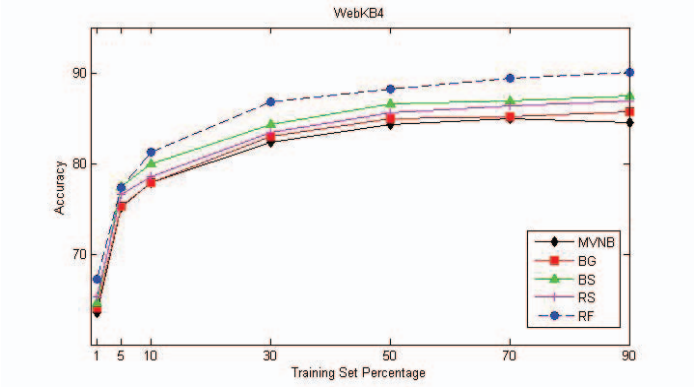


Fig. 3. Performance comparison of the ensemble classifiers on WebKB4 dataset.

Figure 2, 3, 4, 5 represent the performance comparisons of the ensemble algorithms and base learner for all datasets at different training set percentages. In Figure 2 and 3, it is clearly seen that random forest algorithm exceeds others at all training set percentages for 20News-18828 and WebKB4 and base classifier, MVNB, has the worst classification performance and shows similar classification success rate with bagging algorithm for both datasets. While accuracy values are close to each other at smaller training set percentages, the difference between them increases at higher training set levels.

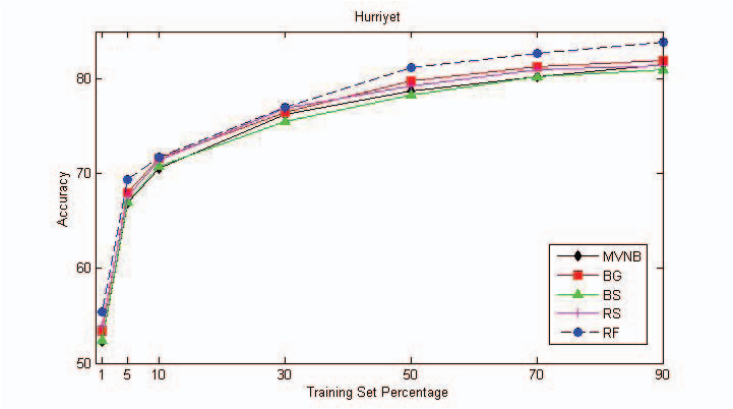


Fig. 4. Performance comparison of the ensemble classifiers on Hurriyet dataset.

At smaller training set sizes, the dominance of the random forest is not observed especially from ts1 to ts10 for Hurriyet and ts1 to ts30 for Aahaber dataset. At remaining training set percentages, the success pattern of the random forest is similar to the other figures. Moreover, the order of classification success rates can change depending on the datasets but it can be observed that the surpassing one is the random forest ensemble algorithm in all cases and the worst classification performance belongs to the base classifier, MVNB.

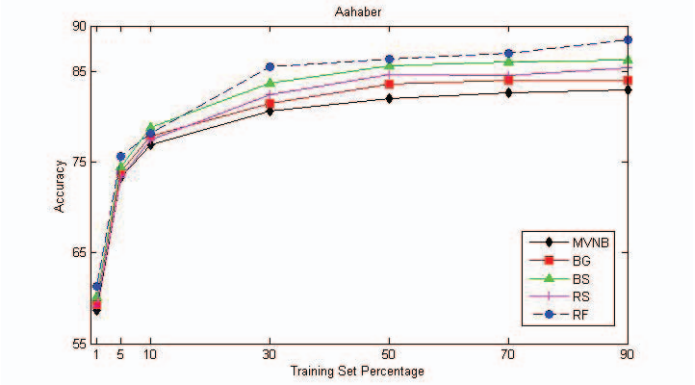


Fig. 5. Performance comparison of the ensemble classifiers on Aahaber dataset.

VI. Discussion and Conclusions

In this study, we attempt to observe the effectiveness of ensemble algorithms on supervised text categorization and investigate the classification performances the two versions of naïve Bayes model, SVM, decision tree classifiers and four homogenous ensemble classifiers that use the multivariate naïve Bayes as a base classifier on four widely-used datasets.

As a first step, the classification performances of different machine learning algorithms (MVNB, MNB, SVM, and DT) are researched to determine the base learner for ensemble system. Due to its outstanding success, MVNB is preferred as a base learner to construct the ensemble of classifiers as mentioned before in section V.

After determining base classifier, ensemble algorithms such as bagging, boosting, random subspace, and random

forest are built by using the base learner. We obtained interesting experimental results which indicate that the preference of bagging and random subspace algorithms as an ensemble technique does not provide significant influence compared to the boosting and random forest algorithms on English text documents. On the other hand, this situation is not usually valid for Turkish documents and these ensemble techniques slightly raise the classification performances for Turkish datasets. Besides, the random forest algorithm is the best ensemble algorithm among others that produces the highest classification performance in text categorization field. Boosting is the next best ensemble algorithm that competes better than BG and RS. Hence, it will be appropriate to identify random forest technique that excels the overall performance of the system. We can summarize that the overall performance order is as follows: RF>BS>RS>BG>=MVNB.

We performed extensive experiments by using different ensemble algorithms and datasets in both English and Turkish. In conclusion, experimental results demonstrate that the application of the ensemble algorithms is an effective technique in order to boost the overall classification success in text categorization. As a future work, we plan to construct heterogeneous ensembles that use different types of base classifiers and investigate the performance of heterogeneous ensemble systems on different datasets.

REFERENCES

[1] Sebastiani, F, “Machine learning in automated text categorization”. ACM Computing Surveys. Vol. 34(1). pp. 1-47, 2002.

[2] Aggarwal, Charu C. and Zhai, ChengXiang. "A Survey of Text Classification Algorithms.." In Mining Text Data , edited by Charu C. Aggarwal and ChengXiang Zhai , 163-222. : Springer, 2012.

[3] L. Rokach, “Ensemble-based classifiers,” Artificial Intelligence Review, vol. 33, no. 1-2, pp. 1–39, 2010. [Online]. Available: http://dx.doi.org/10.1007/s10462-009-9124-7.

[4] R. Polikar, “Ensemble based systems in decision making,” IEEE Circuits and Systems Magazine, vol. 6, no.3, pp. 21-45, 2006.

[5] D. Gopika and B. Azhagusundari, “An analysis on ensemble methods in classification tasks,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 3, no. 7, pp. 7423–7427, 2014.

[6] Y. Ren, L. Zhang and P. N. Suganthan, "Ensemble Classification and Regression-Recent Developments, Applications and Future Directions [Review Article]," in IEEE Computational Intelligence Magazine, vol. 11, no. 1, pp. 41-53, Feb. 2016.

[7] Thorsten Joachims. Text Categorization with Suport Vector Machines: Learning with Many Relevant Features. In Proceedings of the 10th European Conference on Machine Learning (ECML '98), Claire Nedellec and Céline Rouveirol (Eds.). Springer-Verlag, London, UK, UK, 137-14, 1998.

[8] Yan-Shi Dong and Ke-Song Han, "A comparison of several ensemble methods for text categorization," Services Computing, 2004. (SCC 2004). Proceedings. 2004 IEEE International Conference on, 2004, pp. 419-422.

[9] Yan Liu, Jaime G. Carbonell, and Rong Jin. ECML, volume 2837 of Lecture Notes in Computer Science, page 277-288. Springer, (2003).

[10] M. Kanakaraj and R. M. R. Guddeti, "Performance analysis of Ensemble methods on Twitter sentiment analysis using NLP techniques," Semantic Computing (ICSC), 2015 IEEE International Conference on, Anaheim, CA, 2015, pp. 169-170.

[11] Yaxin Bi, David Bell, Hui Wang, Gongde Guo, and Jiwen Guan. 2007. Combining Multiple Classifiers Using Dempster's Rule For Text Categorization. Appl. Artif. Intell. 21, 3 (March 2007), 211-239.

[12] M. J. Gangeh, M. S. Kamel and R. P. W. Duin, "Random Subspace Method in Text Categorization," Pattern Recognition (ICPR), 2010 20th International Conference on, Istanbul, 2010, pp. 2049-2052.

[13] Boros, Martin; Marsík, Jirí. Multi-Label Text Classification via Ensemble Techniques, International Journal of Computer and Communication Engineering1.1 (May 2012): 62.

[14] Haytham Elghazel, Alex Aussem, Ouadie Gharroudi, Wafa Saadaoui, Ensemble multi-label text categorization based on rotation forest and latent semantic indexing, Expert Systems with Applications, Volume 57, 15 September 2016, Pages 1-11.

[15] Aytug Onan , Serdar Korukoglu , Hasan Bulut , Ensemble of keyword extraction methods and classiÞers in text classiÞcation, Expert Systems With Applications (2016).

[16] McCallum A, Nigam KA. Comparison of Event Models for Naive Bayes Text Classification. In: AAAI-98 Workshop on Learning for Text Categorization; 1998; Wisconsin, USA: pp. 41-48.

[17] Christopher J. C. Burges. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. Data Min. Knowl. Discov. 2, 2 (June 1998), 121-167.

[18] L. Rokach and O. Maimon. 2005. Top-down induction of decision trees classifiers - a survey. Trans. Sys. Man Cyber Part C 35, 4 (November 2005), 476-487.

[19] Breiman L. Bagging predictors. Machine Learning, 24(2):123–140, 1996.

[20] Y. Freund and R. E. Schapire, “Experiments with a new boosting algorithm,” in Proc. International Conference on Machine Learning (ICML.96), vol. 96, 1996, pp. 148–156.

[21] T. K. Ho, "Random subspace method for constructing decision forests," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 832-844, 1998.

[22] Panče Panov and Sašo Džeroski. Combining bagging and random subspaces to create better ensembles. In Proceedings of the 7th international conference on Intelligent data analysis (IDA'07), Michael R. Berthold, John Shawe-Taylor, and Nada Lavrač (Eds.). Springer-Verlag, Berlin, Heidelberg, 118-129, 2007.

[23] Breiman, L.: Random forests. Machine Learning. 45(1), 5–32 (2001).

[24] McCallum A, Nigam KA. Comparison of Event Models for Naive Bayes Text Classification. In: AAAI-98 Workshop on Learning for Text Categorization; 1998; Wisconsin, USA: pp. 41-48.

[25] Schneider KM. On Word Frequency Information and Negative Evidence in Naive Bayes Text Classification. In: 4th International Conference on Advances in Natural Language Processing; 2004; Alacant, Spain: pp. 474–485

[26] Rennie JDM, Shih L, Teevan J, Karger DR. Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In: 20th International Conference on Machine Learning; 2003; Washington, USA: pp. 616-623.

[27] Juan A, Ney H. Reversing and Smoothing the Multinomial Naive Bayes Text Classifier. In: 2nd International Workshop on Pattern Recognition in Information Systems; 2002; Alacant, Spain: pp. 200-212.

[28] Kolcz A, Yih W. Raising the Baseline for High-Precision Text Classifiers. In: 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2007; California, USA: pp. 400-409.

[29] Eyheramendy S, Lewis DD, Madigan D. On the Naive Bayes Model for Text Categorization. In: 9th International Workshop on Artificial Intelligence and Statistics; 2003; Key West, Florida, USA: pp. 332-339.

[30] Kim SB, Han KS, Rim HC, Myaeng SH. Some Effective Techniques for Naïve Bayes Text Classification. IEEE Transactions on Knowledge and Data Engineering 2006; 8: 1457-1465.

[31] Vilar D, Ney H, Juan A, Vidal E. Effect of Feature Smoothing Methods in Text Classification Tasks. In: 4th International Workshop Pattern Recognition in Information Systems; 2004; Porto, Portugal: pp. 108-117.

[32] Peng F, Schuurmans D, Wang S. Language and Task Independent Text Categorization with Simple Language Models. In: Human Language Technology Conference; 2003; Edmonton, Canada: pp. 110-117.

[33] Peng F, Schuurmans D. Combining Naïve Bayes and n-Gram Language Models for Text Classification. In: 25th European Conference on Information Retrieval Research; 2003; Pisa, Italy: pp. 335-350.

[34] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA Data Mining Software an Update. ACM SIGKDD Explorations Newsletter 2009; 11.

[35] Kilimci ZH, Ganiz MC. Evaluation of classification models for language processing. In: 10th International Symposium on INnovations in Intelligent SysTems and Applications; 2015; Madrid, Spain: pp. 1-8.

[36] Saha S, Murthy CA, Pal SK. Rough set Based Ensemble Classifier forWeb Page Classification. Fundamenta Informaticae 2007; 76: 171-187.

[37] Amasyalı MF, Ersoy OK. Classifier Ensembles with the Extended Space Forest. IEEE Transactions on Knowledge and Data Engineering 2013; 26: 549-562.

[38] Adnan MN, Islam MZ, Kwan PWH. Extended Space Decision Tree. 13th International Conference on Machine Learning and Cybernetics; 2014; Lanzhou, China: pp. 219-230.

[39] Kilimci ZH, Akyokus S. N-Gram Pattern Recognition using MultivariateBernoulli Model with Smoothing Methods for Text Classification. 24th IEEE Signal Processing and Communications Applications Conference; 2016; Zonguldak, Turkey.

[40] Tantug, AC. Document Categorization with Modified Statistical Language Models for Agglutinative Languages. International Journal of Computational Intelligence Systems 2010; 5: 632-645.