The Evaluation of Heterogeneous Classifier Ensembles for Turkish Texts

Zeynep Hilal Kilimci[1], Selim Akyokus[2]
Computer Engineering Department[1, 2]
Dogus University [1, 2]
Istanbul, 34722, Turkey[1, 2]
{hkilimci, sakyokus@dogus.edu.tr[1,2]}

Sevinç İlhan Omurca
Computer Engineering Department
Kocaeli University
Umuttepe Yerleşkesi, Kocaeli, 41380, Turkey
silhan@kocaeli.edu.tr

***Abstract*— The basic idea behind the classifier ensembles is to use more than one classifier by expecting to improve the overall accuracy. It is known that the classifier ensembles boost the overall classification performance by depending on two factors namely, individual success of the base learners and diversity. One way of providing diversity is to use the same or different type of base learners. When the same type of base learners is used, the diversity is realized by using different training data subsets for each of base classifiers. When different type of base classifiers used to achieve diversity, then ensemble system is called heterogeneous. In this paper, we focus on the heterogonous ensembles that use different types of base learners. An ensemble system based on classification algorithms, naïve Bayes, support vector machine and random forest is used to measure the effectiveness of heterogeneous classifier ensembles by conducting experiments on Turkish texts. Experiment results demonstrate that the usage of heterogeneous ensembles improves classification performance for Turkish texts and encourages to evaluate the impact of heterogeneous ensembles for the other agglutinative languages.**

Keywords— Heterogeneous ensembles; majority voting; stacking; svm; naïve Bayes; random forest; text categorization.

I. INTRODUCTION

Recently, text classification/categorization has attracted the attention of many researchers due to the huge number of documents available on the different digital platforms. The main objective in text categorization to classify a given text or document into a set of predefined categories by using supervised learning algorithms. The supervised learning algorithms learn and generate a model of relationship between features and categories. After the generation of model, the learning algorithm can predict the category of a given document.

A text categorization task consists of parsing of documents, tokenization, stemming, stop-words removal, and representation of documents in document-term matrix with different weighting methods, feature selection, selection of the best classifiers by training and testing [1]. Different methods are used for each of subtasks given above. Each document is represented in vector space model (also called bag of words model). In vector space model, each document is represented as a vector that includes a set of terms (words) that appears in the document. The set of documents with their vector representation forms the document-term matrix. The significance of each term in a document is computed by using different term weighting methods. Term weighting methods include Boolean, term frequency and TF-IDF weighting schemes. There are many types of supervised algorithms (classifiers) used in text categorization. A review and comparison of supervised algorithms is presented in papers [1,2]. Some of the commonly used supervised algorithms include naïve Bayes (NB), k-nearest neighbors (k-NN), decision trees (DT), artificial neural networks (ANN) and support vector machines (SVM).

One of the existing trends in machine learning is to use an ensemble of classifiers to improve the classification performance. In an ensemble system, a group of base classifiers is used. If base classifiers use different learning algorithms, then it is called heterogeneous ensemble; otherwise homogenous ensemble. A heterogeneous ensemble system is composed of two parts: ensemble generation and ensemble integration [3-7]. In ensemble generation part, a diverse set of base classifiers is generated. Naïve Bayes, support vector machine and random forest are used as base classifiers in this study. There are many integration methods that combines decisions of base classifiers to obtain a final decision [8-10]. In this study, we focus on heterogeneous ensembles and utilize majority voting and stacking methods for ensemble integration.

We have evaluated and discussed the performance of heterogeneous four base learners and two integration methods using four different Turkish datasets on this work. This paper is organized as follows: Section 2 gives related research on text categorization using ensemble systems. Section 3 presents base learners and ensemble fusion methods used in experimental studies. Experimental setup and results are given in sections 4 and 5. Section 6 summarizes and discusses results, and outlines future research directions.

II. RELATED WORK

High dimensionality of input feature space, sparsity of document vectors, the presence of few irrelevant features are the main characteristics of text categorization problem [11] that differs from other classification problems. There has been little research to solve text categorization problem using ensemble systems. One of initial work of applying ensemble systems to text categorization is done by Larkey and Croft in [12]. They used the combination of three classifiers, K-nearest-neighbor, Relevance feedback and Bayes classifiers to categorize medical documents. Dong and Han [13] used three different variants of naive Bayes and SVM classifiers. They compared the performance of six different homogenous ensembles and a heterogeneous ensemble classifier. Fung et al. [14] uses a

978-1-5090-5795-5/17/$31.00 ©2017 IEEE

heterogeneous ensemble classifier that uses a dynamic weighting function to combine decisions. A pairwise ensemble approach is presented in [15] that achieve better performance than popular ensemble approaches bagging and ECOC. S. Keretna et al. [16] have worked on named entity recognition problem using an ensemble system.

For Twitter sentiment analysis, an ensemble classifier is proposed in [17] where the dataset includes very short texts. A combination of several polarity classifiers provides an improvement of the base classifiers. In another study done by Gangeh et al [18], Random subspace method is applied to text categorization problem. The paper emphases the estimation of ensemble parameters of size and the dimensionality of each random subspace submitted to the base classifiers. Boros et al [19] 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. [20] proposes a novel ensemble multi-label text categorization algorithm, called Multi Label Rotation Forest (MLRF), based on a combination of Rotation Forest and Latent Semantic Indexing. In a recent study, the predictive performance of ensemble learning methods on text documents that are represented keywords is evaluated by Onan et al [21] empirically. The five different ensemble methods that use four different base classifiers are applied on the documents represented by keywords.

III. Base Learners and Ensemble Integration Techniques

This section briefly reviews learning algorithms and ensemble integration techniques used in experiments. This study uses two variants of Naïve Bayes: MVNB (multivariate Bernoulli naïve Bayes) and MNB (multinomial naïve Bayes), SVM (support vector machines), and RF (Random Forest) learning algorithms as base classifiers to generate a heterogeneous ensemble system. To integrate decisions of base classifiers, majority voting and stacking methods are applied.

A Naïve Bayes classifier is a simple probabilistic classifier based on Bayes' theorem which assumes the independence of features from each other. There are two variants of Naïve Bayes classifier frequently used for text categorization. These two classifiers are called multivariate Bernoulli naïve Bayes (MVNB) and multinomial naïve Bayes (MNB). In MVNB, each document is represented by a vector with binary variables that can takes values 1 or 0 depending upon the presence of a word in the document. In MNB, each document vector is represented by the frequency of words that appears in the document. Equation (1) defines MVNB classifier with Laplace smoothing. The 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. $P(c_j|d_i)$ is 1 if document i is in class j. The probability of term w_t in class c_j is as follows [22]:

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

Support vector machine (SVM) is binary classifier that divides an n-dimensional space with n features into two regions related with two classes [23]. The n-dimensional hyperplane separates two regions in a way that the hyperplane has the largest distance from training vectors of two classes called support vectors. SVM can also be used for a non-linear classification using a method called the kernel trick that implicitly maps input instances into high-dimensional feature spaces that can separated linearly. In SVM, the use of different kernel functions enables the construction of a set of diverse classifiers with different decision boundaries.

Random forests are a collection of decision tree classifiers introduced by Breiman [24]. It is a particular implementation of bagging in which decision trees are used as base classifiers. Given a standard training set, the bagging method generates a new training set by sampling with replacement for each of base classifiers. In standard decision trees, each node of the tree is split by the best feature among all other features using a splitting criteria. To provide randomness at feature space, Random Forest algorithm first selects a random subset of features and then decides on the best split among the randomly selected subset of features. Random forests are strong against to overfitting because of randomness applied in both sample and feature spaces.

To combine decisions of individual base leaners, majority voting and stacking methods is used in this study. In majority voting method, an unlabeled instance is classified according to the class that obtains the highest number of votes from collection of base classifiers. In stacking method, also called stacked generalization, a meta-level classifier is used to combine the decision of base level classifiers [25]. The stacking method consists of two steps. In the first step, a set of base-level classifiers $C_1, C_2, \ldots, C_n$ is generated from a sample training set S that consists of feature examples $s_i = (\mathbf{x}_i, y_i)$ where $\mathbf{x}_i$ is feature vector is and y_i is prediction (class label). A meta-data set is constructed from the decisions of base-level classifiers. The meta-dataset contains an instance for each instance in the original training dataset. The meta-data set can be in the form of $m_i = (\boldsymbol{d}_i, y_i)$ where $\boldsymbol{d}_i$ are predictions of individual n base classifiers. In some problems to improve performance, the meta-data set includes both original training examples and decisions of base level classifiers in the form of $m_i = (\mathbf{x}_i, \boldsymbol{d}_i, y_i)$. After the generation of meta-data set, a meta-level classifier is trained with meta-data set and used to make predictions. In our study, the meta-data set include both original training examples and decisions of base level classifiers.

IV. Experiment Setup

We use four well-accepted benchmark datasets with different sizes and properties to explore the classification performances of the heterogeneous classifier ensembles. Milliyet dataset includes text from columns of Turkish newspaper Milliyet from years 2002 to 2011. It contains nine categories and 1000 documents for each category. The categories of this dataset are café (cafe), dünya (world), ege (region), ekonomi (economy), güncel (current), siyaset (politics), spor (sports), Türkiye (Turkey), yaşam (life).

Hurriyet dataset includes news from 2010 to 2011 on Turkish newspaper Hurriyet. It contains six categories and 1000 documents for each category. Categories in this dataset are dünya (world), ekonomi (economy), güncel (current), spor (sports), siyaset (politics), yaşam (life). 1150haber dataset is obtained from a study done by Amasyalı and Beken [26]. It consists of 1150 Turkish news texts in five classes (economy, magazine, health, politics, sports) and 230 documents for each category.

Aahaber, collected by Tantuğ [27], is a dataset consists of newspaper articles broadcasted by Turkish National News Agency, Anadolu Agency. This dataset includes eight categories and 2,500 documents for each category. Categories are Turkey, world, politics, economics, sports, education science, “culture and art” and “environment and health”. Milliyet and 1150haber include the writings of the column writers therefore they are longer and more formal. On the other hand, Hurriyet and Aahaber datasets contains traditional news articles. They are more irregular, much shorter than documents of the other datasets.

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 only filter infrequent terms whose document frequency is less than three. We don’t apply any stemming or stop word filtering in order to avoid any bias that can be introduced by stemming algorithms or stop-word lists.

TABLE I. CHARACTERISTICS OF THE DATASETS WITH NO PREPROCESSING

Dataset	**|C|**	**|D|**	**|V|**
1150haber	5	1,150	11,040
Milliyet	9	9000	63,371
Hurriyet	6	6,000	18,280
Aahaber	8	20,000	14,395

We perform experiments using 80% training and 20% test data with repeated holdout method. The holdout is applied 10 times on each dataset. This approach is similar to the previous works [28-33] where they use 80% of data for training data and 20% for test. In this study, predicative accuracy, F-measure and AUC (Area Under the Curve) evaluation measures are used to compare results of the experiments. The averages of the evaluation measures are computed along with deviations from the averages.

V. EXPERIMENT RESULTS

At first, we analyze the classification performances of individual classifiers to compare them with each other. The learning algorithms multivariate Bernoulli naïve Bayes (MVNB) and multinomial naïve Bayes (MNB), SVM (support vector machines), and RF (Random Forest) are used as classifiers.

Table II demonstrates the evaluation results of the individual classifiers explained in section III on four datasets: 1150haber, Milliyet, Hurriyet, and Aahaber. Each cell includes the average and deviation from the average by applying 10 repetitions of the holdout method. It clearly seems that RF is more competitive and performs better than other classification techniques. We can summarize that the classification success order of the single classifiers are as follows: RF > MVNB > MNB > SVM.

TABLE II. THE CLASSIFICATION ACCURACIES OF THE SINGLE CLASSIFIERS

Dataset	**MVNB**	**MNB**	**SVM**	**RF**
1150haber	93.74±1.35	94.00±1.64	89.65±2.62	94.32±1.02
Milliyet	84.64±0.95	81.48±1.03	89.41±0.53	90.18±0.87
Hurriyet	81.16±0.96	79.78±0.78	76.58±1.31	84.13±0.96
Aahaber	82.70±1.07	82.80±0.93	79.62±1.12	87.35±1.37
avg	***85.56±1.08***	***84.52±1.09***	***83.82±1.39***	**88.99±1.06**

To construct a heterogeneous ensemble system, we used MVNB, MNB, SVM and RF algorithms as base classifiers. The decisions of each of base classifiers are combined with Majority Voting (MV) and Stacking (STCK) integration methods as described in Section III. In Table III, the classification accuracies of each of the single classifiers, the heterogeneous ensemble with majority voting (Heter-MV) and the heterogeneous ensemble with stacking (Heter-STCK) are compared.

TABLE III. THE CLASSIFICATION ACCURACIES OF THE SINGLE CLASSIFIERS AND HETEROGENEOUS ENSEMBLE ALGORITHMS AT TS80.

Methods	**1150haber**	**Milliyet**	**Hurriyet**	**Aahaber**
MVNB	93.74±1.35	84.64±0.95	81.16±0.96	82.70±1.07
MNB	94.00±1.64	81.48±1.03	79.78±0.78	82.80±0.93
SVM	89.65±2.62	89.41±0.53	76.58±1.31	79.62±1.12
RF	94.32±1.02	90.18±0.87	84.13±0.96	87.35±1.37
Heter-MV	**95.71±0.83**	**92.27±0.76**	**84.08±0.51**	**87.06±0.97**
Heter-Stck	**97.16±1.23**	**94.57±2.06**	**85.44±0.92**	**87.73±1.17**

From Table III, we can observe that the Heter-MV and Heter-STCK ensembles produce better accuracies than single classifiers. The Heter-STCK ensemble demonstrates the superior classification performance among all of the base classifiers and the Heter-MV majority voting ensemble model for each dataset. Heter-MV (Majority voting ensemble model) is competitive in terms of ensemble accuracies and it also shows better classification success compared to the individual classifiers. While Heter-STCK (stacking) reaches maximum 2% success and increment of the classification performance in proportion to majority voting model, classification success of it ranges from 2% to 13% compared to the single classifiers for different datasets. The lowest and highest accuracy improvements differ according to the datasets 3% - %8 (1150haber), 4% - 13% (Milliyet), 1% - 9% (Hurriyet), and maximum 8% increment for Aahaber. Especially for Milliyet dataset, 13% rise provides the significant contribution to the classification performance in terms of the usage of the ensemble strategies. Another observation of this experiment is that the performance results of heterogeneous ensembles Heter-MV and Heter-STCK is better than Random Forest (RF) algorithm. RF is in fact a homogenous ensemble system that uses a collection of decision tree classifiers.

We also try to investigate the performance results of each of the ensemble classifiers in more detail on our four datasets. Accuracy, F-measure and AUC (Area Under the Curve) are utilized as the evaluation criteria in experiments. Table IV shows the performances of Heter-MV ensemble classifier. Majority voting ensemble model achieves 95% (1150haber), 91% (Milliyet), 82% (Hurriyet), and 84% (Aahaber) F-scores on the related datasets. It can't significantly boost the system performance compared to stacking ensemble approach.

TABLE IV. THE CLASSIFICATION RESULTS OF HETER-MV ENSEMBLE ALGORITHM.

Dataset	Accuracy	F-Measure	AUC
1150haber	95.71±0.83	95.82±1.03	99.18±0.22
Milliyet	92.27±0.76	91.62±0.53	98.76±0.84
Hurriyet	84.08±0.51	83.75±0.96	99.82±0.09
Aahaber	87.06±0.97	85.90±0.41	97.66±0.34

Table V shows the performances of Heter-STCK stacking ensemble approach. The stacking ensemble approach exhibits F-scores of 97% (1150haber), 94% (Milliyet), 84%(Hurriyet), 85% (Aahaber). Some of the AUC (Area Under the Curve) measures is near to one. The AUC is suitable for evaluating the performance of classifiers on unbalanced data sets. The classifier with high AUC values maximizes the true positive rate while minimizing the false positive rate. It is obvious that Heter-STCK has the superior performance among all other methods.

TABLE V. THE CLASSIFICATION RESULTS OF HETER-STCK ENSEMBLE ALGORITHM.

Dataset	Accuracy	F-Measure	AUC
1150haber	97.16±1.23	97.32±0.82	99.23±0.05
Milliyet	94.57±2.06	94.03±1.25	99.21±0.18
Hurriyet	85.44±0.92	85.57±1.10	99.73±0.27
Aahaber	87.73±1.17	86.81±0.72	98.45±0.71

VI. DISCUSSION AND CONCLUSIONS

In this paper, we focus on heterogeneous ensembles by using different types of classification algorithms and ensembles integration approaches. The success of an ensemble system depends on the diversity of the base classifiers and methods that make up the ensemble system. The different learning algorithms, namely variants of two Naïve Bayes, support vector machine and random forest are chosen to provide diversity for the ensemble system. The majority voting and stacking ensemble integration strategies are applied in order to consolidate the final decision of the ensemble system. In this study, a wide range of comparative and extensive empirical studies are conducted on four widely-used datasets in Turkish. In this way, we try to measure the effectiveness of heterogeneous classifier ensembles by conducting experiments on Turkish texts. The results of experiments noticeably demonstrate that the usage of heterogeneous ensembles improve the classification performances on Turkish texts and encourages the researchers to evaluate the impact of heterogeneous ensembles on the other agglutinative languages.

The classification performance of individual classifiers changes depending on the algorithm and dataset. It appears that random forest algorithm is more competitive and outperforms other classification techniques. We can summarize the classification success order of the single classifiers like this: RF > MVNB > MNB > SVM. In our ensemble system, two type of integration strategies are used to combine the decision of individual classifiers. The majority voting integration strategy achieves the significant improvements over single classifiers, yet it can't considerably exceed the system performance of the stacking integration approach. Therefore, the best ensemble integration approach is the stacking method that provides the best performance results in our experiments.

In an earlier study [33], authors focus on the effectiveness of homogenous ensemble algorithms on English and Turkish texts and investigate the success of two versions of naïve Bayes model, SVM, decision tree classifiers and four homogenous ensemble techniques that use the multivariate naïve Bayes as a base classifier. It is observed that homogenous ensemble approach is an effective technique to improve the overall classification success. In this study, we try to measure the effectiveness of ensemble techniques using heterogeneous ensemble models. The average accuracies of heterogeneous ensembles with 89.8% (majority voting) and 91.2% (stacking) exceed the best homogenous ensemble result 88.1% at the same training set percentages for all datasets. Although all datasets are chosen Turkish texts in this study, there are two common Turkish datasets namely, Hurriyet and Aahaber with the study [33] in order to compare the classification results of systems. While the overall average accuracy of two datasets is 85.7% for the homogenous ensemble model (HE), accuracies of heterogeneous ensemble strategy for majority voting (HEM) and stacking (HES) are 85.6% and 86.6%, respectively. When HES slightly improves the success of HE result, the accuracy results of HE and HEM are close to each other at the same training set percentage for two datasets.

In conclusion, it is observed that the use of heterogeneous ensemble strategy is an efficient technique to enhance the classification performance of the system on Turkish texts. In addition, heterogeneous ensemble systems provide slight improvements over homogenous ensemble systems. As a future work, we will try to boost the performances of classifier ensembles further by using novel models.

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] Reid, Sam. "A review of heterogeneous ensemble methods." Department of Computer Science, University of Colorado at Boulder (2007).

[6] 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.

[7] 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.

[8] Mangai, U. G., Samanta, S., Das, S., & Chowdhury, P. R. (2010). A survey of decision fusion and feature fusion strategies for pattern classification. IETE Technical review, 27(4), 293-307.

[9] Woźniak, M., Graña, M., & Corchado, E. (2014). A survey of multiple classifier systems as hybrid systems. Information Fusion, 16, 3-17.

[10] Tsoumakas, G., Angelis, L., & Vlahavas, I. (2005). Selective fusion of heterogeneous classifiers. Intelligent Data Analysis, 9(6), 511-525.

[11] Thorsten Joachims. Text Categorization with Support 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.

[12] Larkey, Leah S., and W. Bruce Croft. "Combining classifiers in text categorization." In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 289-297. ACM, 1996.

[13] 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.

[14] Liu, Yan, Jaime Carbonell, and Rong Jin. "A new pairwise ensemble approach for text classification." In European Conference on Machine Learning, pp. 277-288. Springer Berlin Heidelberg, 2003.

[15] Fung, Gabriel Pui Cheong, Jeffrey Xu Yu, Haixun Wang, David W. Cheung, and Huan Liu. "A balanced ensemble approach to weighting classifiers for text classification." In Data Mining, 2006. ICDM'06. Sixth International Conference on, pp. 869-873. IEEE, 2006.

[16] Keretna, Sara, Chee Peng Lim, Doug Creighton, and Khaled Bashir Shaban. "Classification ensemble to improve medical named entity recognition." In Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on, pp. 2630-2636. IEEE, 2014.

[17] 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.

[18] 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.

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

[20] 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.

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

[22] 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.

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

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

[25] Džeroski, Saso, and Bernard Ženko. "Is combining classifiers with stacking better than selecting the best one?." Machine learning 54, no. 3 (2004): 255-273.

[26] Amasyalı, M. Fatih, and Aytunç Beken. "Türkçe kelimelerin anlamsal benzerliklerinin ölçülmesi ve metin sınıflandırmada kullanılması." In IEEE signal processing and communications applications conference, SİU-2009. 2009.

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

[28] 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.

[29] 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.

[30] 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.

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

[32] 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.

[33] Kilimci, Z. H., Selim Akyokus, and Sevinc Ilhan Omurca. "The effectiveness of homogenous ensemble classifiers for Turkish and English texts." In INnovations in Intelligent SysTems and Applications (INISTA), 2016 International Symposium on, pp. 1-7. IEEE, 2016.