Available online at: http://brsj.cepsbasra.edu.iq

ISSN -1817 -2695

THE COMBINATION OF TEXT CLASSIFICATION SYSTEM

[1] ZAINAB M. JAWAD, [2] ZAINAB A. KHALAF

[1] *Basrah, College of Education for Pure Sciences, Computer Science Dept.*
[2] *Basrah University, College of Science, Mathematics Science Dept.*

E-mail: [1] z82m89asia@gmail.com, [2] zainab_ali2004@yahoo.com

ABSTRACT

Text classification is considered an important task for arranging text data under labels or categories. Real life data varies in language, size, format, noise and area. Consequently, these data need to be classified accurately due to their sensitivity and importance. Despite there being different classifiers that could be used, any individual classifier is still affected by issues and does not produce the best result for all data. Therefore, the researchers directed to use a system combination to enhance the prediction result. In this paper, a combination system is used to avoid the variation in the classifiers' performance, by finding the best prediction among three classifiers (Naïve Bayes, Support Vector Machine, and K-Nearest Neighbors) before giving the final decision, to achieve the required reliability and the efficiency. Four data collections are used in English and Arabic languages: 20-newsgroup, Reuters-21578, Watan-2004 and Khalaf-2018. The proposed system produces better results than the individual classifiers. The F1-scores for the proposed system were 95%, 93%, 94% and 92% for 20-newsgroup, Reuters-21578, Watan-2004 and Khalaf-2018, respectively.

Keywords: *Text Classification, Combination system, Naïve Bayes, Support Vector Machine, K-Nearest Neighbors.*

1. INTRODUCTION

With the explosive growth of the Internet and the increasing amount of text data, the web has become a huge repository that is full of information. Billions of web pages exist containing all kinds of knowledge stored in textual format within different sources such as articles, books, literature, papers, E-mails and messages [1]. The textual data and the information are stored in an unstructured format. Therefore, with the enormous growth and complexity of the unstructured textual data, there is a pressing need to arrange this data under labels or categories to improve the efficient use of information; this can be

done by text classification [1][2]. Text classification is the assigning of a label to an unseen document based on predefined classes [3][4], the text classification will decrease the amount of the data being accessed, optimizing computation and reducing the time taken to search for required information. Many types of supervised machine learning are used for text classification, such as Support Vector Machine (SVM), Naïve Bayes (NB), K-Nearest Neighbors (KNN) and Random Forest (RF) [2].

The main objective of this paper is to improve the classifier results using a post-combination system (multi-classifiers system) by combining the results of three classifiers: NB, SVM and KNN. Two types of combination are used: majority voting and soft combination.

The remaining sections of this paper are structured as follows: Section (2) is a review of the related works. Section (3) discusses the combination system. In Section (4), the proposed system is presented. The experiment results will be discussed in Section (5). The results are discussed in Section (6). Finally, the conclusion is shown in Section (7).

2. RELATED WORKS

Researchers needed to do a lot of work to improve the text classification, either by reducing the features or enhancing the classifiers' performance. Many issues affect the performance result of the classification such as the amount of the data collected, the high dimensionality of features or which classifier was used. Some of the researchers combined low performance classification systems to create a high performance one. Badawi et al [1] used the prior-combination to combine the terms and termsets' features. Also, Wan et al [2] combined features extracted from three different feature extraction methods: SABigram, bi-gram and 2-termset, SABigram that based on tokenized the documents by punctuations and stop words, the bi-gram to extract the features that appear sequentially, 2-termsets are assumed, meaning that two terms appear in one document. In addition, they combined two feature-reduction methods: the chi-square and the relevance category frequency, which performed better with NB than the SVM. Their system would prefer the small vocabulary datasets that have few noisy features. For enhancing the performance of the classifiers, Rahman et al [3] used a during-combination to combine two classifiers: the Multinomial Naive Bayesian (MNB) with the Bayesian Networks (BN) classifier, by averaging the probability distributions of both classifiers. Xu et al [4] used a combination of the multinomial NB classifier and Bayesian called a BayesianNB. The proposed system produced a similar result to the multinomial NB when using 20-Newsgroup data collection. For post-combination, Liu et al [5] used three classification algorithms: NB, KNN and logistic regression. Their proposed system used TF-IDF and Word2vec for feature weighting. The stacking combination used passed the result through two layers of prediction then used cross validation to reach the prediction. Next, they trained the prediction to the second layer to reduce the error. They implemented different stacking and achieved better performance from the individual classifiers. Ondusko et al [6] combined the classifiers Logistic Regression, Multinomial Naive Bayes, Random Forest, SVM, and Perceptron., using a dataset from Kaggle for Twitter. They combined a subset of the classifiers. Each classifier scores every test and sums the score up to 1, then ranks the results. Their proposed system shows good performance for all the classifiers or a

subset of them. Finally, Khalaf et al [7] used both the prior-combination, using the N-gram filtering, and the post-combination using the voting, to enhance the SVM and cosine classifier results. The drawbacks of their system were caused by using only the SVM classifier for the N-gram approach. In the current study, four data collections are used for experiments that varied in language, the size of the data collection, the tools used, and the number of classes. Thereby, different sizes of vocabulary are examined. In addition, many comparisons with the most widely used classifiers and previous related studies that used the same datasets will be shown.

3. COMBINATION SYSTEM

The combination system could be applied in different positions within the classification system. Therefore, the combination system can be divided into three types: prior-combination, during-combination and post-combination [8].

The *prior-combination*: in this type, the combination approach is performed outside the classification system, either by combining more than one data collection, or in a preprocessing step such as the stemming step, or by a composite feature consisting of two or more features to get more informative features than would normally be available from an individual [9].

The *during-combination*: This occurs inside the classifier. They can be implemented hierarchically, serially or in parallel [9].

The *post-combination*: this type combines the result generated by different classifiers by using the hard-combination or the soft-combination to get the best performance for the classification [2][7]**.**

Each classifier can produce two outputs: the prediction that represents the class label, and the estimated score of the correct actual label [8]. Depending on the two outputs of the combination system, the *hard-combination* is based on the prediction values by finding the majority voting, while the *soft-combination* is based on the scores by using the *argmax* functions [8]. The soft-combination consider better than the Hard-combination because of reducing the error rate and enhance the performance of the classification. [10]

4. THE PROPOSED SYSTEM

Text classification is the process of automatically assigning a class label to the new document based on a pre-defined pattern created from labelled documents. The classification system divides the data into training and testing phases. In the training phase, parameters are obtained that are used in the testing phase to predict the label for the new documents [2][11][12]. Different classifiers produce different results, so the main motivation of this study is to improve the individual classifier results by combining muti-classifiers to extract the best decision among the individual classifiers results.

The proposed text classification system consists of four main phases, as shown in figure (1) and explained in the following subsections:

4.1 Preprocessing Phase

Preprocessing is the preparation of the data to facilitate easy handling by the classification algorithms. This phase contains many stages as follows: *Tokenization* is used to separate the documents into words, terms or tokens [13]. *Normalization* is used to remove the symbols and the characters, and unify the multi-shape of the letters that have more than one shape, e.g., in English the letters converted to lower case, and in Arabic some of the letters with multi-shape, such as "ا, إ, آ, أ" are convert to a single shape:

"ا" [13]. *Stop Words* is the list of words that frequently appear in the documents. These words are important to the meaning of the document but are irrelevant to the classification, therefore these words are removed [14]. *Stemming and Lemmatization* is the process of removing the prefixes and the suffixes to convert the word to its root form [14].

The extracted features are represented by using *Vector Space Machine* (VSM), where each document is represented by a vector of frequency to the corresponding feature, with the length of the features that are extracted from this stage. Each feature in the vector is weighted by using TF-IDF.

4.2 Classification Phase

The VSM from previous steps is passed into the classification system. Three classifiers are used: NB, SVM and KNN. Each classifier produces the best hypothesis that can be obtained. However, each classifier generates different errors. For example, suppose these classifiers produced different hypothesis results, H_1: local news, H_2: sport news and H_3: local news. The results are based on the strength of the algorithm used, the knowledge derived from training data, and the way this is used to extract features. Therefore, the post-combination system is applied to take advantage of the strength of different classifiers and generates the most suitable result.

Taking the number of classifiers as M, and the output for the classifiers to one testing document is $[L_1, L_2, \dots, L_M]$, where L is the label predicted from each classifier. The majority voting depends on the frequency of each prediction. If the frequency (*freq*) of the class label is more than half of the number of classifiers, then it will be the final label for the corresponding test document.

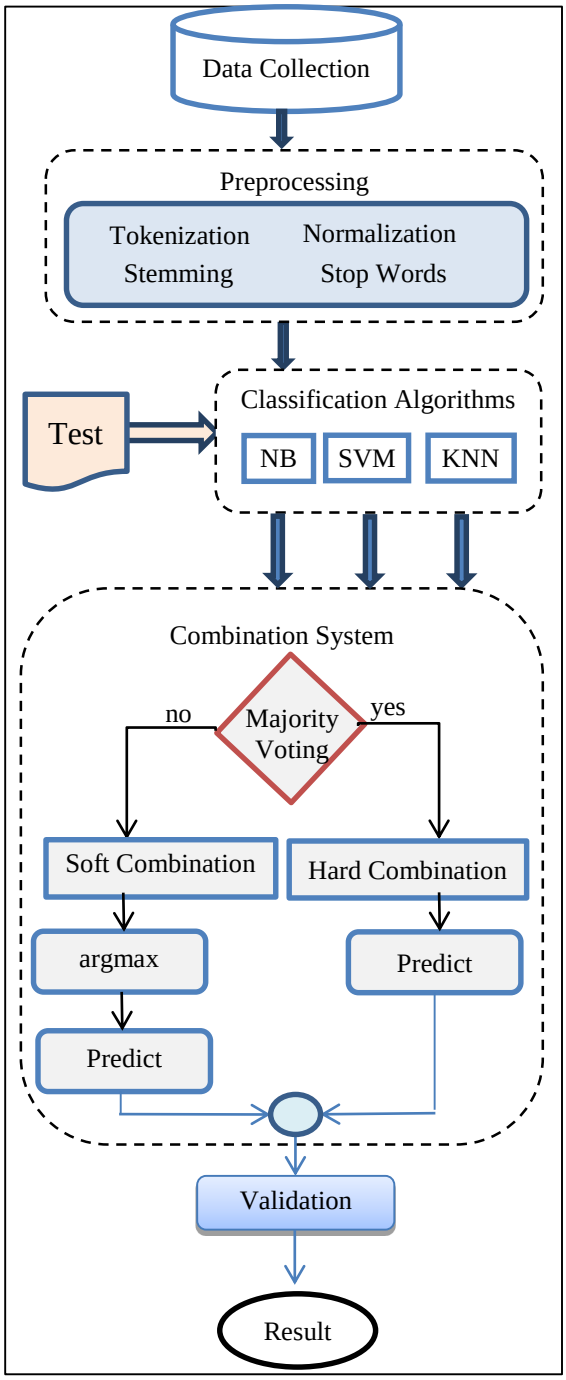

Figure 1: The proposed system architecture

If the majority voting condition is not met, then the soft combination will apply. The soft combination depends on the probability list (*prob*) for every classifier by using the *argmax* function. The *argmax* finds the largest value to the summation of the classes' probabilities.

The equation (1) expresses the combination process for a single testing data.

$$label = \begin{cases} L_i & if\ freq(L_i) > \frac{M}{2} \\ L_j & \forall j \in N, argmax(prob_i, \dots, prob_M)_j \end{cases} \quad (1)$$

Where M is the number of the classifiers, N is the number of the classes, and $i = [1, \dots, M], j = [1, \dots, N]$.

In the above example, suppose the label "local news" occurs twice in the total of the three combined hypotheses. In contrast, the label "sport news" occurs once for the same test document. Therefore, the best voting process is for "local news". In some cases, the classifiers produce different results (labels) by using hard-combination, so the soft-combination will be used to find the best probability instead of labels to determine the best result.

4.3 Evaluation Phase

The performance of the classification is evaluated by using the F1-score that is calculated using the weighted average of the precision and the recall [15].

Precision is the number of documents that are correctly classified divided by the number of documents that are predicted by the system: $Precision = \frac{TP}{TP+FP}$.

Recall is the number of documents that are correctly classified divided by the number of the actual documents in the data: $Recall = \frac{TP}{TP+FN}$.

The precision and recall are contradictory; an improvement in the precision will reduce that of the recall, and vice versa. Therefore, they are integrated into F1-score by the following equation [15]:

$$F1 - score = \frac{2 * (Recall \times Precision)}{(Recall + Precision)} \quad (2)$$

5. EXPERIMENTAL RESULTS

This section describes the experimental performance of the conventional approaches that are frequently implemented in text classification systems. First, the preprocessing approach is applied, and then the stemming and the Part Of Speech (POS), that selects the nouns, verbs and adjectives, is used to extract the features. The new features obtained are fed to the classifiers as input. Three methods are used as baselines: NB, SVM and KNN from the sklearn package. The parameters for the classifiers are:
NB: MultinomialNB (alpha = 1.0).
SVM: SVC (kernel = "linear").
KNN: KNeighborsClassifier (k = 5).

The validation is done using these classifiers as baselines, also after the combination. The current experiment is applied by Python 3.7 using a computer with an Intel Core i7-8550U CPU, 16GB RAM and 64-bit processor.

5.1 Data Collections

Four data collections across two languages (English and Arabic) are used in this study. For English, 20-Newsgroups [16] and Reuters-21578 [17] data collection are used, and for Arabic, Watan-2004 [18] and Khalaf-2018 [7]. The 20-Newsgroups consisted of 19,997 documents that were classified into 20 classes. Each class comprised approximately 1,000 files. The Reuters-21578 corpus was comprised of 135 classes, of which the top seven were used. These included 10,806 documents. The Watan-2004 data collection contained around 20,291 documents distributed into six classes. Khalaf-2018, the second Arabic corpus, was collected manually by [7] in 2018 from the online websites of

BBC and CNN Arabic news from June to December 2018. This corpus consisted of 2,750 Arabic news documents distributed into six classes. The corpora used in this study were divided into 75% documents for training and 25% for testing.

5.2 The Classifiers Performance

The classifier results are examined in this section. First, the preprocessing is applied. The preprocessing is an essential step for every data collection before applying the classification step. After the tokenization, normalization and remove stop-list steps are applied to the datasets, the Stanford corenlp lemmatizer is used for both languages. Porter-Stemmer and ArabicLight-Stemme are used for English and Arabic, respectively.

The features extracted from the preprocessing step for the four data collections are shown in table (1):

Table 1: The Number of features for the data collections

Data collection	No. of features
20-newsgroup	86151
Reuters-21578	18431
Watan-2004	78130
Khalaf-2018	31980

After the preprocessing phase, the features extracted from each data collection are weighted using TF-IDF and then passed to the classifiers. In the following subsection, baseline and proposed system results are shown.

5.2.1 The Results of the Baselines

This section describes the experimental performance of the conventional approaches that are frequently implemented in text classification. Three methods that are used as baselines are applied to the databases: Linear NB, SVM and KNN. After applying the preprocessing phase, the features extracted from each database are passed to the classifiers. The main aim of the NB algorithm is to find the probability result that determines the most likely class. Meanwhile, the main aim of linear SVM is to find the best hyperplane separator between the document classes. The idea of the KNN is to find the K nearest neighbors depending on the distance between the documents. Table (2) shows the performance of the baseline classification.

Table 2: The baseline results

	20-newsgroup	Reuters-21578	Watan-2004	Khalaf-2018
NB	90%	72%	86%	71%
SVM	95%	92%	93%	91%
KNN	80%	83%	92%	88%

5.2.2 The Proposed System Results

This section examines the performance of the proposed system that applies the post-combination to combine the results of classifiers. The results of the NB, SVM and KNN are combined and then the highest voting or similarity is selected to find the most appropriate class. The performance of combination system is 95%, 93%, 94% and 92% for 20-newsgroup, Reuters-21578, Watan-2004 and Khalaf-2018, respectively.

6. DISCUSSIONS

The proposed system aims to enhance the performance of the classification by combining multi-classifiers. The combination system was applied to the data collections using three classifiers: NB, SVM, and KNN. The baseline results were enhanced to (95%, 93%, 94% and 92%) for 20-Newsgroup, Reuters-21578, Watan-2004 and Khalaf-2018, respectively.

In addition, the proposed system achieved better results compared to the related works.

The researchers Wan et al. [2] used the prior-combination by combining features extracted from three different methods.

These methods are: SABigram, bi-gram and 2-termset, using 20-newsgroup and 10 classes from Reuters-21578. NB and SVM were used as classifiers, and the F1-scores were 64% and 82% respectively, for 20-newsgroup. For Reuters-21578, the F1-scores for NB and SVM were 56% and 93%, respectively. In the current study, Stanford lemmatizer and Porter Stemmer were used and the classifier result was better.

In the Khalaf-2018 data collection that was collected by Khalaf et al. [7], the F1-score result for SVM was 90.9%, while in the proposed system the F1-score result for SVM was 91.4%. The researchers used khoja stemmer, while Stanford corenlp lemmatizer and ArabicLightStemme were used in the current study. Table (3) shows other researchers' works using different combination systems.

Table 3: Comparisons with Related Works

Reference	The Combination Tools	The Combination Type	The Classifiers Model	The Data Collection	No. of Classes	F1-Score
[1] 2018	Terms and termset features	Prior	SVM	20-newsgroup - 19997	20	78%
[2] 2019	The feature extracted (bigram, 2-termset and SABigram)	Prior	SVM, NB	20-newsgroup-19997	20	82%, 64%
				Reuters-2157	10	93%, 56%
[4] 2019	Multinomial NB classifier and Bayesian	During	BayesianNB	20-newsgroup-18828	20	91%
[7] 2018	N-gram, the classifiers Cosine and SVM	Prior + Post	Cosine, SVM	Khalaf-2018	6	81%, 91%
Current Study	NB, SVM and KNN	Post	NB, SVM, KNN	20-Newsgroup, Reuters-21578, Watan-2004, Khalaf-2018	20 7 6 6	95% 93% 94% 92%

7. CONCLUSION

In text classification, the most important issues are the efficiency and reliability of the classification. The researchers tried to improve the performance of the classifiers using different methods. In this study, we present a system to combine three widely used classifiers to improve the performance of the classification: NB, SVM, and KNN. The combination system aims to find the best label and consequently enhance the performance result. The combination results were (95%, 93%, 94% and 92%) for 20-Newsgroup, Reuters-21578, Watan-2004 and Khalaf-2018, respectively. The proposed system enhances the classification performance for the classifiers results individually.

In future work, many combination approaches could be used at the feature extraction or preprocessing phases, or by using a confidence score to find the best predication.

REFERENCES

[1] Badawi Dima and Altınçay Hakan, "Compact Representation Of Documents Using Terms And Termsets," *In: Perner P. (eds) Machine Learning and Data Mining in Pattern Recognition. Lecture Notes in Computer Science*, **10934**, 77, (2018).

[2] Wan Chuan, Wang Yuling, Liu Yaoze, Ji Jinchao & Feng, Guozhong, "Composite Feature Extraction And Selection For Text Classification", *IEEE Access*, **7**, 1, (2019).

[3] Rahman Amna and Qamar Usman, "A Bayesian Classifiers Based Combination Model For Automatic Text Classification," *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences*,63, (2017).

[4] Xu Shuo, Li Yan & Wang Zheng, "Bayesian Multinomial Naïve Bayes Classifier To Text Classification," *Lecture Notes in Electrical Engineering, Springer Nature Singapore*, **518**, **15**, (2019).

[5] Liu Jun, Shang Wenqian & Lin Weiguo, "Improved Stacking Model Fusion Based On Weak Classifier And Word2vec," *IEEE- International Conference on Computer and Information Science in Singapore*, 820, (2018).

[6] Ondusko Dominik, Ho James, Roy Brandon and Hsu Frank, "Sentiment Analysis On Tweets Using Machine Learning And Combinatorial Fusion," *IEEE Intelligence Conference on Dependable, Autonomic and Secure Computing, Intelligence Conference on Pervasive Intelligence and Computing, Intelligence Conference on Cloud and Big Data Computing, Intelligence Conference on Cyber Science and Technology, Fukuoka, Japan*, **1**, 1066, (2019).

[7] Khalaf Zainab A. and Hassan Khadeija A., "Filtering Approach And System Combination For Arabic News Classification," *Journal of Theoretical and Applied Information Technology*, **96**, **14**, 4491, (2018).

[8] Mohandes Mohamed, Deriche Mohamed and Aliyu Salihu, "Classifiers Combination Techniques : A Comprehensive Review", *IEEE Access*, **6**, 19626, (2018).

[9] Fierrez Julian, Morales Aythami, Vera-Rodriguez Ruben and Camacho David, "Multiple classifiers in biometrics. part 1: Fundamentals and review", *Information Fusion, Elsevier*, **44**, 57, (2018).

[10] Toufiq Rizoan and Islam Rabiul, "Face Recognition System Using Soft-output Classifier Fusion Method,", *2nd International Conference on Electrical, Computer & Telecommunication Engineering*, 8, (2016).

[11] M. Anandarajan, C. Hill, and T. Nolan, "Practical Text Analytics," *Anandarajan M., Hill C., Nolan T. Classif. Anal. Mach. Learn. Appl. to Text. Pract. Text Anal. Adv. Anal. Data Sci. vol 2. Springer, Cham*, **2**, 131, (2019).

[12] Irfani Fakhruddin Farid, Fauzi M. Ali and Sari Yuita Arum, "News Classification on Twitter Using Naive Bayes and Hypernym-Hyponym Based Feature Expansion,", *International Conference on Sustainable Information Engineering and Technology (SIET), Malang, Indonesia*, 317, (2018).

[13] Kobayashi Vladimer & Mol Stefan & Berkers Hannah & Kismihók Gabor & Hartog Den, "Text Classification For Organizational Researchers: A Tutorial", *Organizational Research Methods*, **21**, **3**, 766, (2018).

[14] Kadhim Ammar, "Survey On Supervised Machine Learning Techniques For Automatic Text Classification", *Springer Artificial Intelligence Review*, **52**, 273, (2019).

[15] Yin Chunyong and Xi Jinwen, "Maximum Entropy Model For Mobile Text Classification In Cloud Computing Using Improved Information Gain Algorithm,", *Springer, Multimedia Tools and Applications*, **76**, **16**, 16875, (2017).

[16]http://qwone.com/~jason/20Newsgroups.

[17] http://konect.cc/networks/gottron-reuters .

[18]https://sourceforge.net/projects/arabiccorpus/

نظام الدمج في تصنيف النصوص

[1] زينب مهدي محمد جواد, [2] د. زينب علي خلف
[1]جامعة البصرة, كلية التربية للعلوم الصرفة, قسم الحاسبات
[2]جامعة الصرة, كلية العلوم, قسم الرياضيات

E-mail: [1] z82m89asia@gmail.com, [2] zainab_ali2004@yahoo.com

الملخص

يعتبر تصنيف النصوص من المهمات الضرورية لترتيب البيانات النصية ضمن فئات او اصناف. تختلف البيانات الحقيقية من ناحية اللغة والحجم والتنظيم والضوضاء و المجالات التي تهتم بيها. وبالتالي، يجب ان تصنف هذه البيانات بدقة نظرا لحساسيتها وأهميتها. على الرغم من وجود مصنفات مختلفة يمكن استخدامها، إلا ان التصنيف الفردي لايزال متأثرا ببعض المشكلات ولا يعطي افضل نتيجة لجميع البيانات. لذلك، توجه الباحثون إلى استخدام نظام دمج المصنفات الفردية لتعزيز نتيجة التصنيف. في هذا البحث، سيتم استخدام نظام الدمج لتجنب التباين في اداء المصنفات الفردية من خلال البحث عن افضل تنبؤ بين المصنفات الثلاثة (Naïve Bayes , Support Vector Machine و K-Nearest Neighbors) قبل اتخاذ القرار النهائي، لتحقيق الموثوقية والكفاءة المطلوبة. في هذه الدراسة تم استخدام اربع مجاميع بيانات باللغتين الانكليزية والعربية 20-newsgroup، Reuters-21578، Watan-2004 و Khalaf-2018. حقق نظام الدمج المقترح نتائج افضل من النتائج المصنف الفردي، وكانت نتيجة F1-score هي 95%، 93%، 94% و 92% لكل من 20-newsgroup و Reuters-21578 و Watan-2004 و Khalaf-2018 على التوالي.

الكلمات المفتاحية: تصنيف النصوص، نظام الدمج، Naïve Bayes ، Support Vector Machine و K-Nearest Neighbors

.