FUZZY LOGIC BASED COMPATIBLE SUPPORT VECTOR MACHINE (FL-CSVM) FOR TEXT DOCUMENT CLASSIFICATION

Dr.E Chandra Blessie , Deepa A
Professor, Ph.D Research Scholar
Department of Computer Applications
Nehru College Of Management, Coimbatore, India.

Abstract: Document classification using keyword extraction has become an important domain for researchers in text mining. The main intention of keyword extraction is to represent the document in a precise manner. The compatible details of documents acts as a multiple applications in different ways. Currently, the document classification based on keywords has become a main task. Most available classifiers are well suited only for the dataset which have minimum number of documents. In this paper, Fuzzy Logic Based Compatible Support Vector Machine (FL-CSVM) is proposed to classify the text documents in maximum accuracy. FL-CCSVM is designed to perform classification by dividing the documents dataset into multiple parts in a random manner, where the previous classification algorithm follows sequential manner classification leading to poor classification accuracy. The existing classifiers gives better results only when the dataset is small, but FL-CSVM is designed to accept the dataset with any size to give increased classification accuracy. To evaluate the proposed classifiers performance against previous classifiers this work chooses three text document collection dataset namely ACM Document collection dataset, Reuters-21578, and NBA Input document collection dataset holding 3506, 21578, and 1256 documents respectively. The results shows that FL-CSVM is having better performance in terms of Classification Accuracy and F-Measure, than baseline classifiers.

Index Terms: Classification, Mining, Text, NBA, ACM.

1. INTRODUCTION

Text mining research has attracted many researchers due to the incredible quantity of text data being generated in day to day daily life in social media, records of patients in hospitals, insurance data, news channel and so on. It was predicted that the volume of text data in the year 2020 might reach 4.0E+13 gigabytes, which will be fifty times of text data volume in the year 2010. Unstructured information's best example is considered as text data only, where it is considered as the easiest way to create in many scenarios. Humans can perceive and process the unstructured text in easy manner, but the same is notable very harder to understand by the machines. Text mining is as a part of data mining and methods of knowledge discovery, but with certain specificities. Text classification is considered as important part text mining, where text documents are assigned classes that are predefined. These predefined classes makes easy way to manage and sort the documents. Most of the current applications are based on text mining, that is related to research problem of text classification. Traditional classifier simply classify the documents based on control structure oriented conditions in first in first out order basis, that is in a sequential basis.

1.1 Problem Statement and Motivation

Day by day misclassification issue is keep on increasing due to the increased dimensional feature space. While utilizing the complete set of words available in training documents for feature selection, the text classification process becomes exhaustive task by means of computations. The exhaustive task leads to wasting the time of human as well as the CPU. Hence there arise a need for better classifier to identify keywords collection, which will responsible to identify contents of the document in a more accurate manner in a less delay.

The reminder of this paper is organized with Literature Review as Section 2, Proposed Work as Section 3, Keyword Extraction methods gets discussed in Section 4, Regarding the Datasets in Section 5, Evaluation Measure as Section 6, Experimental Results as Section 7, and Section 8 concludes the paper with its future dimensions.

2. LITERATURE REVIEW

Hidden Markov-Model based Text Classification [1] proposed for analyzing the sentiments. It utilizes the words in a sequence manner for training, where the other classifiers uses sentiment lexicon that are previously defined. It attempted to learn the patterns and it performs classification. The results indicate that the classifier has low performance over online feedbacks. Piece Wise Linearity Classification [2] aimed to perform the classification based on text frame recognition. It detects the text with the assumption that all the inputs received were text, which was based on component linearity. In the initial stage, text clusters were received in a gradient manners then it was extracted for processing. The gradient manner of receiving the text becomes a disadvantage in this method and it reduced the classification accuracy. Categorized Classification [3] was proposed to categorize the text in the field of construction field. It measures the method of evaluating correct candidate in the available category. The classifier was tested against the two better alternative candidate category, but the true positive values came with very low values. Fisher Discriminant Analyzing Method [4] was proposed for classifying the Arabic texts. In this the authors have used the corpus that has multiple thousand documents and attempted to find the linear based transformation that increases the class segregation rate. The results indicate that the dimensionality method has few effect over Arabic text classification. Ontology based Classification [5] was proposed to classify the health related text data. The health data were transformed to structured data from unstructured data. Multiple domain data were focused to extract the relevant information. It utilizes the technical related terms and attempted to enhance it, but the results shows that the algorithm has poor performance towards classification.

Self Training based Classification [6] was proposed to perform classification using semi-supervised methods and representation models. It determined the settings of the parameter from document collection. The concept of self training is utilized to enhance the labeled data. The classification accuracy got reduced when the algorithm was investigated with topic based representation. Semi Supervised Clustering [7] attempted to increase the accuracy of text classification, the main idea was to segregate a category of one cluster of text into multiple clusters. The labeled texts were utilized to grab the outline of the text clusters and unlabeled texts were utilized to grab the outline of centroids. The results indicate that the algorithm has consumed more time and returned poor classification accuracy. k-NN and Naïve-Bayes Classification Methods [8] has focused to perform the classification by ensembling the 2 classification methods. The data sets holds 20 different types of XML documents, where 6 types of documents were selected. The increased false positive rate represents the algorithms weakness and it reflected in classification accuracy.
Linguistic Features based Classification [9] was proposed to detect the plagiarism in a binary manner. It involves the process of distinguishing the text either as non-plagiarized or plagiarized. The classification was performed in the intermediate stage. Further, dimensionality reduction was also done to increase the accuracy. The results with low accuracy shows the algorithm has poor performance over the classification process. Dimensionality reduction based Classification [10] aimed to reduce the dimension of the text dataset, it lead to have heavy computation because of size of data. It utilizes the hidden markov model combined with k-NN and SVM classification algorithms. The accuracy of the algorithm due to heterogeneous data in the dataset, because the k-NN and SVM classification algorithms supports the homogeneous data.

Orthographic Error Tagging based Classification [11] was proposed to evaluate the spelling error in the text oriented datasets. The algorithm was applied in longitudinal study to categorize the important child's error profiles. The classification result with low accuracy indicates that the algorithm is having low performance over the considered dataset. Semantic Matching based Text Classification [12] was proposed to classify the text contents by defining the selection rules in order to find the relevant information. Similarity between the documents were computed to enhance the classification accuracy by the results reached the failure level due to applying the more rules. Index based Sentiment Classification [13] was proposed to identify the sentiments in various domains by measuring the index of sentiments. It utilized the lexical elements in a single domain's independent features. It attempted to analyze the polarity of short text by utilizing the features on unlabeled data. The results demonstrated that the algorithm cannot improve the efficiency in cross-domain sentiment classification. Multi Text Classification [14] was proposed to perform classification on dynamic and static clusters in an iterated manner. Symmetry method was used to identify the width of stroke. The potential text were grouped by utilizing the properties of geometry in order to form the appropriate text regions. The results demonstrate that the algorithm has medium accuracy and it consumed more time period. Improved Gaussian based Text Classification [15] was proposed to reduce the dimension of the dataset and perform classification. Map reducing concept is further utilized to face the increased level classification. The results of map reducing was ensemble with kNN to increase the classification accuracy, but the final results indicate that the algorithm has lower effects on automatic text classification.

Random Forest [18] was a hybrid algorithm, which makes use the concepts of classification and regression trees. In this algorithm, simplifying the error of the classifier was dependent on the two things, the power of the (i) individual trees, and (ii) association between trees. Features were selected in the random manner, which results in the decrease of accuracy. Bagging Random Forest [19] algorithm was proposed with the focus of building a robust classifier with increased performance towards prediction by merging the classification algorithms on various training sets. To increase the accuracy, random sampling with replacement concept was used.

3. FUZZY LOGIC BASED COMPATIBLE SUPPORT VECTOR MACHINE (FL-CSVM)

The major concept of SVM is to decrease the error that arise in classification when it preserves the increased margin among the classes. If the task of binary classification tends with samples $\{xs\} = 1 \in Rd$ with equivalent labels $z^t \in (-1,1)$, then SVM intend to find the hyperplane with (i) $(x^S \times w^t) + a = 0$, which satisfies the Eqn. (1)

$$z^t(x^S \times w^t) + a = 1, \text{ for all t} \quad (1)$$

where x vector belongs to $\mathbb{Z}^c$ and the preference term a belongs to $\mathbb{Z}$. The man intention of this condition is to make sure that all the samples were at the maximum distance from the hyperplane.
The parameter variables were received by resolving the primal issues through constrained optimization issues and it is mathematically expressed as:

$$e(w) = |x|/2 \quad (2)$$

Eqn. (2) is subject to $z^t(x^S\, w^t + a) + 1 = 0$. In Eqn. (1) , $|x| = x^{i+1} + \cdots, x^{c+2}$ and it indicates the normalization of the vector x. To minimize the complex level, the double penetration trick is utilized to initiate the convex optimization in Eqn. (1).

SVM seeks a hyperplane to classify the samples. The core concept of CSVM is to find two hyperplanes in the available samples that are allocated to a class based on the distance that are from the hyperplanes, it is mathematically expressed as:

$$\begin{cases} x^{S+1}\, w^t + a^{i+1} = 0 \\ x^{S+2}\, w^t + a^{i+2} = 0 \end{cases} \quad (3)$$

where x^j and a^j represents the variable parameters of the hyperplane j. The hyperplanes are not in parallel, and it is expected to have the samples in close to class that in opposite direction.

Consider a task of binary classification that have $+1$ & -1 classes, and B belongs to $\mathbb{Z}^{m^{i+1}*c}$ and A belongs to $\mathbb{Z}^{m^{i+2}*c}$. It denotes the matrix samples that belong to class $+1$ & -1, appropriately. Every matrix rows indicates a sample that belong to the appropriate class. The 2-hyperplanes of CSVM are received by resolving the Eqn. (4) and Eqn.(5).

$$\min \frac{1}{2}(Bx^{i+1} + fa^{i+1})^S(Bx^{i+1} + fa^{i+1}) + o^{i+1}f^S\,\omega \quad (4)$$

Eqn.(4) is subject to $(Ax^{i+1} + fa^{i+1}) + \omega = f, \omega = 0$

$$\min \frac{1}{2}(Ax^{i+2} + fa^{i+2})^S(Ax^{i+2} + ea^{i+2}) + o^{i+2}f^S\,\omega \quad (5)$$

Eqn.(5) is subject to $Bx^{i+2} + fa^{i+2} + \omega = f, \omega = 0$.

In Eqn.(4) and Eqn.(5) the term ω indicates the vector of limp variables having the size m, $\omega = 0$ indicates the component of the vector that is having value as non-negative. Suppose the samples that are used for training are not separable in linear, then general approach is followed to take decision on margins in order mistakes. Every non-zero element of limp variables describes the cost of misclassification samples that are related to measure if distance between margin samples, and the samples of true decision. In Eqn.(4) and Eqn.(5), o^{i+1} and o^{i+2} indicates the parameter of penalty.

CSVM is a binary form of classifier as it incorporates the concepts of SVM. It transforms the inequality conditions used in SVM to equalize the conditions and resolves it to binary linear systems rather than changing it to quadratic program based issues. Its main objective is to minimize the time used for training, which results in increasing the classification accuracy. The traditional SVM's time complexity have the order m^{i+3}, where m represents the count of constraints. In theory, count of samples in first class and second class in the task of binary classification are almost equal. When the count of samples in the first class and second class are equal, then CSVM performs classification 2 times speeder than traditional SVM.

CSVM obtains the disentanglement hyperplanes by using the optimization functions and it mathematically expressed as:

$$\min \frac{1}{2}\,(Bx^{i+1} + fa^{i+1})^S(Bx^{i+1} + fa^{i+1}) + \frac{o^{i+1}}{2}\omega^S\omega \quad (6)$$

Eqn. (6) is subject to $(Ax^{i+1} + fa^{i+1}) + \omega = f$

$$\min \frac{1}{2}(Ax^{i+2} + fa^{i+2})^S(Ax^{i+2} + fa^{i+2}) + \frac{o^{i+2}}{2}\omega^S\,\omega \quad (7)$$

Eqn. (7) is subject to $(Bx^{i+2} + fa^{i+2}) + \omega = f$

By solving the Eqn. (4) and Eqn. (5) CSVM will receive the parameters of hyperplanes (i.e. x and a), as Eqn. (8) and Eqn. (9)

$$\begin{bmatrix} x^{i+1} \\ a^{i+1} \end{bmatrix} = (E^S E + 1/o^{i+1}\,F^s F)^{i+1}E^S f \quad (8)$$

$$\begin{bmatrix} x^{i+2} \\ a^{i+2} \end{bmatrix} = (F^S F + 1/o^{i+2}E^S E)^{-1}F^S f \quad (9)$$

Currently, in real time applications samples that are used for training the data does not belong to the specific class. But, in few applications, it is popular to have various important degrees for training the samples. Considering the ambiguity in assigning important values, the fuzzy logic provides a neat way to deal the problem. Fuzzy-Membership-Degree (μs) is possible to define as sample s in training data. The membership-degree is a numeric value that lies between 0 and 1 which describes the degree of a sample that belongs to specific class. Hence, sample that is used for training with membership-degree of μs and it fit to the class $+1$ by ρ^t and fit to the class -1 by $(1 - \rho^t)$.

In FL-CSVM, membership function of fuzzy is describes to allocate μs to samples, where the outliers and noises obtain minimum values. The main intention of FL-CSVM is to establish 2 hyperplanes that differentiate the target classes. In order to achieve the goal, Eqn. (6) and Eqn. (7) are updated as below to show the +1 and negative −1 (i.e., positive and negative) class.

$$\min I^{i+1} = \frac{1}{2}(Bx^{i+1} + fa^{i+1})^S(Bx^{i+1} + fa^{i+1}) + \frac{o^{i+1}}{2}\rho^{S+1}\omega^2 \quad (10)$$

Eqn. (10) is subject to $(Ax^{i+1} + fa^{i+1}) + \omega = f$.

$$\min I^2 = \frac{1}{2}(ax^{i+2} + fa^{i+2})^S(Ax^{i+2} + fa^{i+2})^S(Ax^{i+2} + fa^{i+2})\frac{o^{i+1}}{2}\rho^{S+2}\omega^2 \quad (11)$$

Eqn. (11) is subject to $(Bx^{i+2} + fa^{i+2}) + \omega = f$

In Eqn.(10) and Eqn. (11) ρ^j, j belongs to (1,2), which is the membership vector. $\frac{o^j}{2}\rho^S\omega^{i+2}$ provides the quantity of cost needed for every decision margin. By reorganizing the constraint equations, it is possible to find the constraints for FL-CSVM. It is mathematically expressed as

$$\min I^{i+1} = \left|\frac{1}{2}Bx^{i+1} + fa^{i+1} + \frac{o^{i+1}}{2}\right|, \left|\rho^{i+1}, Ax^{i+1} + fa^{i+1} + f\right| \quad (12)$$

$$\min I^{i+2} = \left|\frac{1}{2}Ax^{i+2} + fa^{i+2} + \frac{o^{i+2}}{2}, \rho^{i+2}\right|, \left|Bx^{i+2} + fa^{i+2} + f\right| \quad (13)$$

The parameters of Eqn. (12) and Eqn. (13) are $x^j, a^j = j + 1$. On solving the above optimization functions (i.e., Eqn. (12) and Eqn. (13)), we calculate the parameters as

$$\begin{pmatrix} x^{i+1} \\ a^{i+1} \end{pmatrix} = \begin{pmatrix} \rho^{i+1}A^S A + \frac{1}{o^{i+1}B^S B} & \rho^{i+1}A^S f + \frac{1}{o^{i+1}B^S f} \\ \rho^{i+1}f^S A + \frac{1}{o^{i+1}f^S B} & \rho^{i+1}m^{i+2} + \frac{1}{o^{i+1}m^{i+1}f} \end{pmatrix} \begin{pmatrix} -A^S f \\ -m^{i+2} \end{pmatrix} \quad (14)$$

$$\begin{pmatrix} x^{i+2} \\ a^{i+2} \end{pmatrix} = \begin{pmatrix} \rho^{i+2}B^S B + \frac{1}{o^{i+1}A^S A} & \rho^{i+2}B^S f + \frac{1}{o^{i+2}A^S f} \\ \rho^{i+2}f^S B + \frac{1}{o^{i+2}f^S A} & \rho^{i+2}m^{i+1} + \frac{1}{o^{i+2}m^{1+2}f} \end{pmatrix} \begin{pmatrix} -B^S f \\ -n^{i+1} \end{pmatrix} \quad (15)$$

Once when the results of the parameters are received, a brand new sample is classified but it is based on the distance from the hyperplane of the corresponding class.

4. KEYWORD EXTRACTION METHODS

4.1 Cooccurrence Statistical Information (CSI)

This method is based on statistics [20]. It aims to provide priority to the significant terms by means of checking the repeated word occurrences in similar sentences. In the initial stage, most repeated words are found, then by checking the same sentences are continued. The words that are repeated are utilized to find the significance of exact term in the document.

4.2 Eccentricity Based (EB)

This method is based on graph drawing [21]. It utilizes the method of vertex centrality for solving the problems in the keyword extraction. In this method, documents are mentioned as labeled graphs, namely undirected graph and edge graph. Documents are treated as vertices. By using the above assumptions, the maximum relevant documents receives the central position in the graph. The central position is utilized to measure the most exact keywords in the documents.

4.3 Most Frequent (MF)

This method [22] is used to seek the repeated terms in the text documents. For seeking, it utilizes the keywords. In order to mean the text documents, matrix structure is used. In this, count of repetition of words is utilized. In this manner, repetition counting gets maximized.

4.4 Term Frequency Inverse Sentence Frequency (TFISF)

This method [23] is based on statistics. It is the enhancement of frequency inverse document-frequency-method. It measures the size of text document sentences. Every sentences available in the text document are treated as a separate vector. The frequency are measured by multiplying the inverse of its sentence.

4.5 Text Ranking (TR)

This method [24] is based on graph model which is used for manipulating the text for processing. The concept is used in many real time tasks for processing natural languages. It is utilized to find the vertices that has more value. In order to extract the keywords, texts and other parts of the sentences are tokenized. Syntax based filter is applied on all tokens to generate a graph. Ranking model is applied to identify the score values of each word.

4.6 Grammar based Reduction of Term Frequency (GRBTF):

This method [25] focus to seek the grammar words used in the text documents which could not be considered as keywords. Further, it cannot eliminate the words from the text documents for ease identification of considered keywords. It involves the process of reading the document, forming the sentence in a matrix manner, initializing the number of terms which equals 0.

5 ABOUT DATASETS

5.1 ACM Document Collection Dataset

ACM document collection dataset [16] consists of 8 sub-dataset, where each sub-dataset holds 5 classes. Its description is provided in Table 1. Thorough experiments are carried out on ACM document collection dataset for evaluating the performance of existing classifier and the proposed classifier.

Table 1 Descriptions of ACM Document Collection Dataset

Col.	Class #	Docs.	Col.	Class #	Docs.
ACM- 1	3D technologies	91	ACM- 5	Tangible and embedded interaction	81
	Visualization	72		Management of data	96
	Wireless mobile multimedia	82		User interface software and technology	104
	Solid and physical modeling	74		Information technology education	87
	Software engineering	82		Theory of computing	103
ACM- 2	Rationality and knowledge	86	ACM- 6	Computational geometry	89
	Simulation	84		Access control models and technologies	90
	Software reusability	72		Computational molecular biology	71
	Virtual reality	83		Parallel programming	96
	Web intelligence	86		Integrated circuits and system design	93
ACM- 3	Computer architecture education	78	ACM- 7	Database systems	104
	Networking and communications systems	75		Declarative programming	101
	Privacy in the electronic society	98		Parallel and distributed simulation	98
	Software and performance	81		Mobile systems, applications and services	95
	Web information and data management	92		Network and system support for games	73
ACM- 4	Embedded networked sensor systems	50	ACM- 8	Mobile ad hoc networking and computing	90
	Information retrieval	71		Knowledge discovery and data mining	105
	Parallel algorithms and architectures	98		Embedded systems	102
	Volume visualization	104		Hypertext and hypermedia	93
	Web accessibility	71		Microarchitecture	105

.

5.2 Reuters-21578 Document Collection Dataset

Reuters-21578 Document Collection Dataset holds ten classes of ModApte Split [17] belonging to Reuters-21578. The essential information concerning the quantity of training and testing samples are provided in Table 2.

Table 2 Descriptions of Reuters-21578 Document Collection Dataset

Label of the Class	Training Samples count	Testing Samples count
Acq	1650	0719
Corn	0181	0056
Crude	0389	0189
Earn	2877	1087
Grain	0433	0149
Interest	0347	0131
Money-fx	0538	0179
Ship	0197	0089
Trade	0369	0117
Wheat	0212	0071

5.3 NBA Input Document Collection Dataset

NBA input criteria document collection dataset consists of 8 sub-criterias, where each sub-criteria holds different classes. Its description is provided in Table 3. NBA input document collection dataset is processed with various keywords for the distinct terms with the methods of statistical keyword extraction. Thorough experiments are carried out on NBA document collection dataset for evaluating the performance of existing classifier and the proposed classifier.

Table 3. Descriptions of NBA Input Document Collection

Col.	Class#	Docs	Col.	Class#	Docs
NBA1_ Student	Select_Proc	26	NBA5_ Library	Books	30
	Stud_Intake_Capacity	30		E-Journals	10
	Enrol_Proc	32		Online Databases	11
	Admn_Process	26		Films_videos	17
	Admn_Guidelines	30		Lib_Mgmt_S/W	20
	Final Result	13		SS_Field Work	14
NBA2_ Faculty	S-F Strength	24		Working_Hours	25
	F_S_R	13		Users_Feedback	13
	R_FT_PTF	18		Inter_Lib_Network	17
	F-Qual	16	NBA6_ Global_ Input	N_IN_Collaborations	30
	F_Retention	20		NIN_AC_Partnerships	16
	Resrch_Proc_Fac	12		NIN_Strategic_Alliance	35
	Fac_Exposure	23		NIN_Exchg_prgms	27
	FDP_Observation	17		NIN_Corp_partners	26
	Out_Exp_CA	22		Resrch_Collaborations	23
	Resrch_Apt_Fac	12	NBA7_ Quality_ Assurance_Policy	Legacy_BSchool_QA	36
NBA3_ Physical_ Infrastructure	Nat_Geo_Access	40		IA_Process_EDU	26
	Dist_Loc	39		CC_Review_process	34
	Phys_Ambience	36		Emp_Orgs_Feedback	28
	Ava_resources	42		APP_Real_CC	33
NBA4_ IT_ Infrastructure	Operating_ICT	12	NBA8_ Finance	Fund_Effectiveness	18
	Use_Ins_Kits	30		Fin_Self_Suff	24
	H/W_S/W-State	26		Fin_Prf_3Yrs	22
	IT_Lab_Usage	18		IFCRS	22
	Wifi_Use	23		PI_Staff	23
	Video_conferencing	23		Scope_Range_FS	22
	Learn_Platforms	25		Ensr_Accountability	26

6. EVALUATION MEASURES

The evaluations of the experiment are done on a personal computer with the configurations of Intel Core i7 processor having speed of 3.40 GHz, and random access memory of 8 gigabytes. The experiments are performed with MATLAB version R2013a.

To measure the prediction performance of existing and proposed classification algorithms, this research work utilizes the traditional performance metrics classification accuracy and F-measure for the evaluation purpose.

- **Classification Accuracy :** Percentage of true values (positives and negatives) against the overall number of instances, which is denoted as Eqn. (16)

$$Classification\ Accuracy = \frac{TP + TN}{(FP + FN + TP + TN)} \tag{16}$$

where TP and TN denotes True positive and True Negative. FP and FN denotes False Positive and False Negative.

- **Precision :** Percentage of true positives over the total of false positives and true positives, which is denoted as Eqn. (17)

$$Precision = \frac{TP}{(FP + TP)} \tag{17}$$

- **Recall :** Percentage of true positives over the total of false negatives and true positives, which is denoted as Eq. (18)

$$Recall = \frac{TP}{(FN + TP)} \tag{18}$$

- **F-Measure :** Percentage of precision and recalls harmonic mean, which is denoted as Eq. (19)

$$F - Measure = \frac{2 \times (Recall \times Precision)}{(Recall + Precision)} \tag{19}$$

7. EXPERIMENTAL RESULTS

In Fig. 1 to Fig. 6, the keyword extraction methods (Co-occurrence Statistical Information (CSI) [20], Eccentricity based Keyword Extraction (EB) [21], Most Frequent based Keyword Extraction Method (MF) [22], Term Frequency Inverse Sentence Frequency (TF-ISF) [23], Text Rank Algorithm (TR) [24]) and Grammar based Reduction of Term Frequency (GRBTF) [25] are plotted in x-axis and percentages are plotted in y-axis. are the keyword extraction methods used. The percentages indicate the output of classification algorithms (Random Forest (RF) [18], Bagging Random Forest (BRF) [19], FL-CSVM [Proposed]. Classification algorithms are combined with keywords extraction methods to measure the effectiveness towards classifying the documents in ACM [16] and Reuters-21578 [17] document collection dataset. Fig. 1 and Fig. 4 shows evaluation result of classification algorithms on ACM [16] dataset. Fig. 2 and Fig. 5 shows evaluation result of classification algorithms on Reuters-

21578 [17] dataset. Fig 3 and Fig. 6 shows the evaluation result of classification algorithms on NBA Input Document Collection Dataset.

7.1 Classification Accuracy Analysis

Classification Accuracy denotes the percentage of documents that are correctly classified based on the keywords. From the Fig. 1 it is evident that proposed classifier FL-CSVM is giving the good performance with all the chosen keywords extraction methods in ACM dataset [16], where the dataset holds 3506 documents. It is to be noted that classification algorithms combined with CSI [20] are giving a very low performance in terms of accuracy towards classifying the documents. The result shows that FL-CSVM (proposed) gives the top level accuracy in all keyword extraction methods EB [21], MF [22], TF-ISF [23], TR [24] and specifically with GRBTF [25]. This is due to the segregating the documents in an indiscriminate manner and performing the classification, where RF and Bagging RF makes classification in a sequential manner. Also, RF [18] and Bagging RF [19] classifier was proposed to support the documents which are fully text-based, where FL-CSVM is proposed to classify the documents even though multimedia (i.e., image) contents are present.

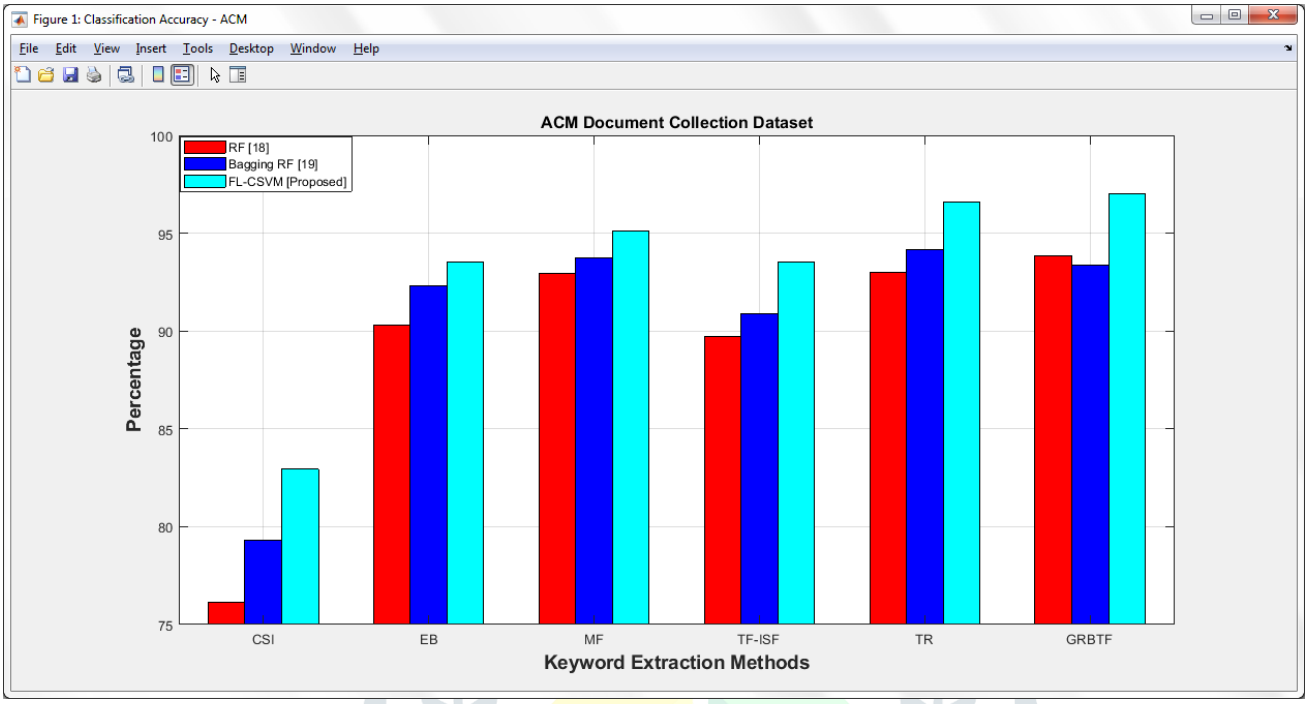
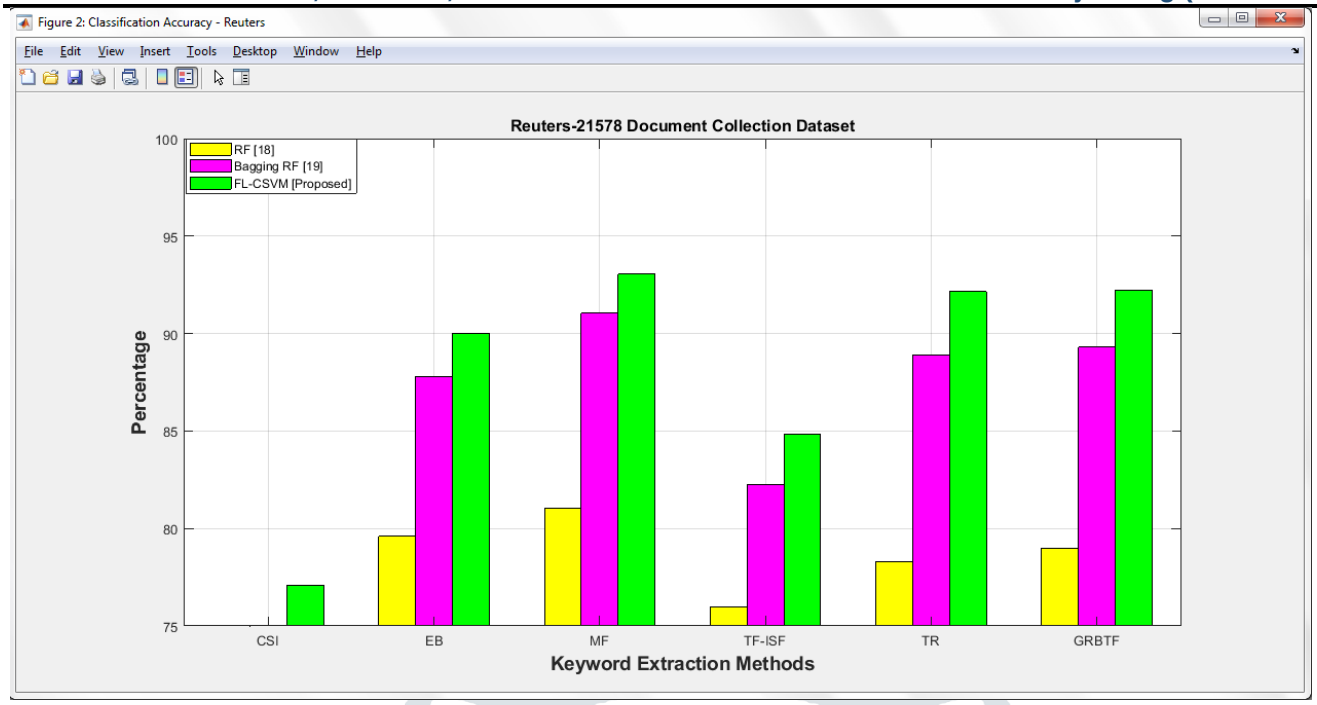


Figure. 1 Classification Accuracy vs ACM Document Collection Dataset

Fig. 2 shows evaluation result of classification algorithms with chosen keyword extraction methods on Reuters-21578 [17] dataset, where the dataset holds 21578 documents. It is clear that FL-CSVM classifier is giving better accuracy when combined with keywords extraction methods, and it to be noted that all classification algorithms combined with CSI [20] is giving low accuracy when compared with other methods namely EB [21], MF [22], TF-ISF [23], TR [24], and GRBTF [25]. It is evident that FL-CSVM is giving better classification result when combined with GRBTF [25], this is due to setting the threshold value for processing the documents. FL-CSVM does not take all the documents at once and process for classification, instead it sets the threshold value and takes the documents for processing in a batch. Due to this, the classification accuracy is increased. The results shows that existing classification algorithms RF [18] and Bagging RF [19] are not fit for huge dataset like reuters-21578 [17], the time taken for classifying gets delayed too much due to following the sequential manner classification.



. 2 Classification Accuracy vs Reuters-21578 Document Collection Dataset

Fig. 3 shows evaluation result of classification algorithms with chosen keyword extraction methods on NBA Input Document Collection Dataset, where the dataset holds 1256 documents. It is clear to see that that the proposed classifier is giving the better accuracy when it is combined with keywords extraction methods. Also RF-CFI [18] gives the poor accuracy due to making the classification in one to one manner. It is evident that FL-CSVM is giving better classification result when combined with GRBTF [25], this is due to utilizing the threshold value concept for processing the documents. Because of using the threshold value, the documents are clustered in a random manner with appropriate groups. Hence the classification accuracy gets increased due to this. The results shows that existing classification algorithms RF [18] and Bagging RF [19] are not fit for NBA Input Document Collection Dataset, the time taken for classifying gets delayed too much due to following the sequential manner classification

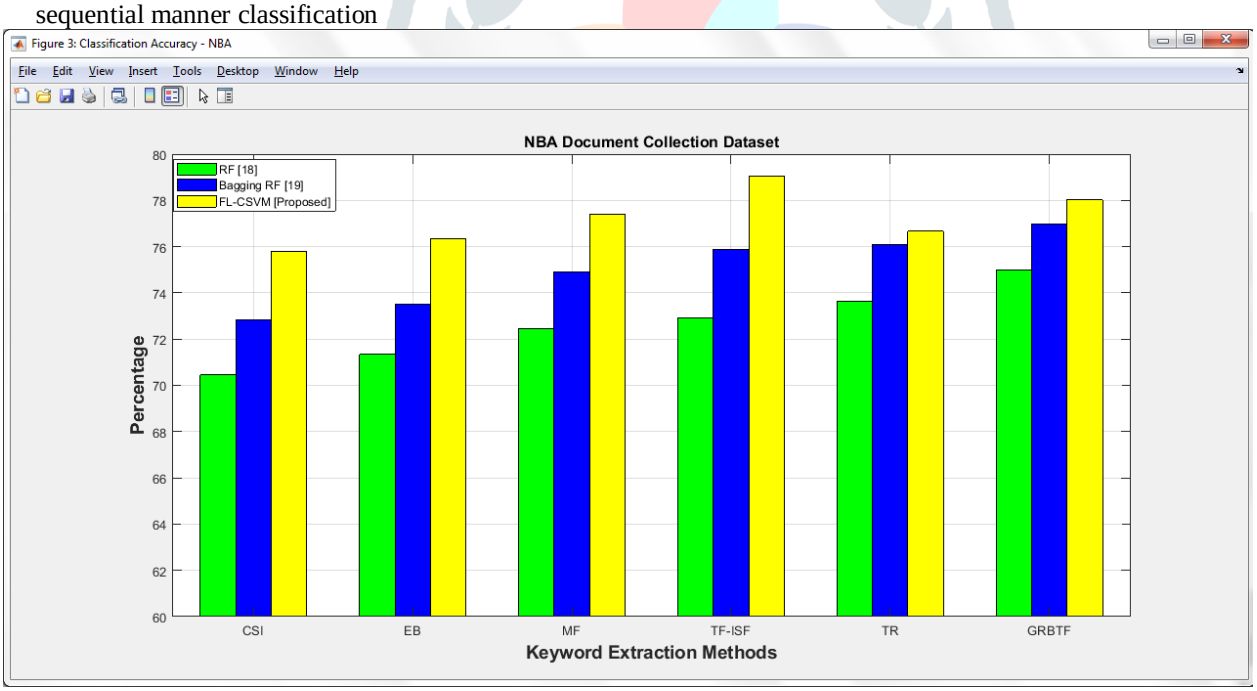


Figure. 3 Classification Accuracy vs NBA Document Collection Dataset

7.2 F-Measure Analysis

F-Measure denotes the percentage of harmonic mean of recall and precision. From Fig. 4 it is noticeable that proposed classifier FL-CSVM haven given remarkable performance with ACM dataset [16], where the dataset holds 3506 documents. The classification algorithms combined with CSI [20] have very low F-Measure. The result shows that the proposed classifier FL-CSVM is able to give best performance with all the keyword extraction methods (EB [21], MF [22], TF-ISF [23], TR [24], GRBTF [25]), where RF [18] and Bagging RF [19] classifiers have given the low performance when comparing with the proposed classifier. The proposed classifier FL-CSVM is having the best performance in precision and recall, where both are used in the calculation of F-Measure. F-Measure of the proposed classier indicates that FL-CSVM is better result in terms of precision and recall.

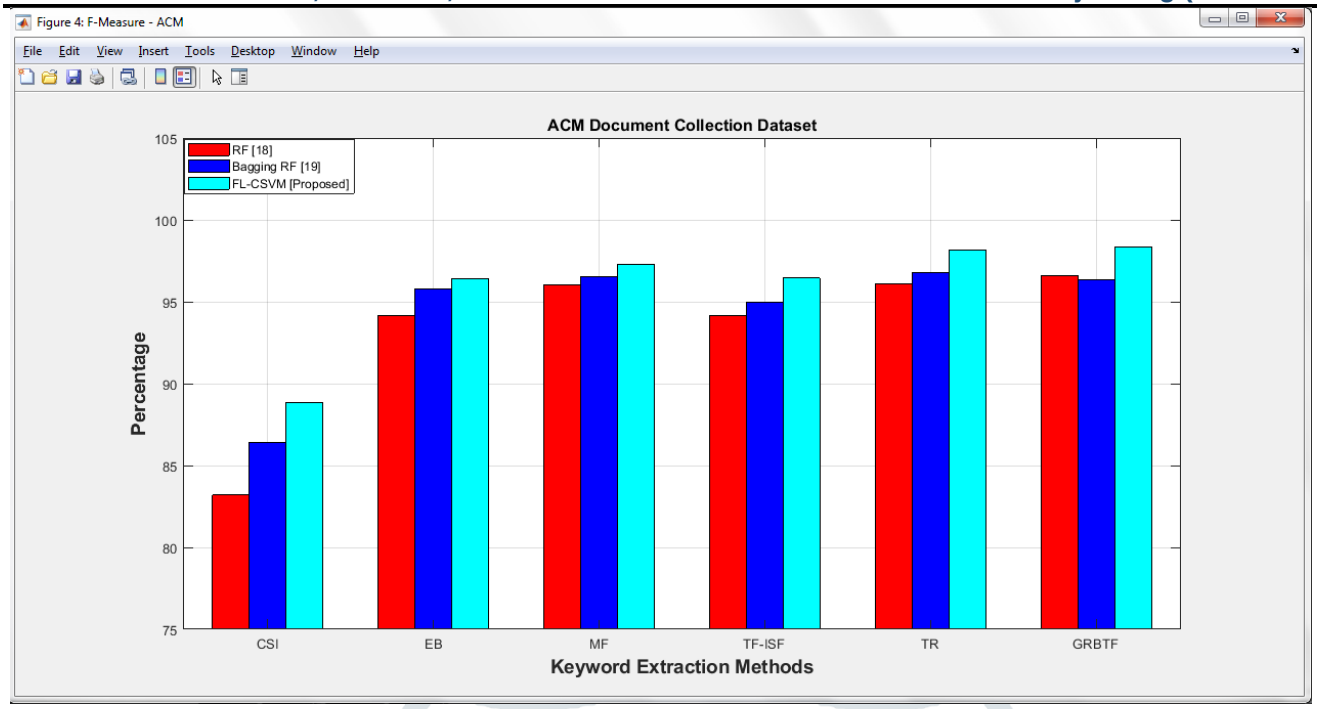


Figure. 4 F-Measure vs ACM Document Collection Dataset

Fig. 5 highlights the F-Measure results of classification algorithms RF [18] and Bagging RF [19] and the proposed classifier FL-CSVM. Fig. 5 clearly demonstrate that the proposed classifier FL-CSVM have given the better F-Measure result with all keyword extraction methods (CSI [20], EB [21], MF [22], TF-ISF [23], TR [24] and specifically with GRBTF [25]), where it has given the low F-Measure with CSI [20]. When analyzing the results, it is found that the keywords extracted by CSI [20] is not sufficient to classify the documents, but FL-CSVM is able to give highest F-Measure when comparing with RF [18] and Bagging RF [19]. The reason for the best outcome of F-Measure by FL-CSVM is it does not take all the documents at once processing the classification, instead it sets the threshold value and takes the documents for processing in a batch. The results shows that FL-CSVM is best suitable for huge dataset like Reuters-21578 [17], when it is combined with GRBTF [25].

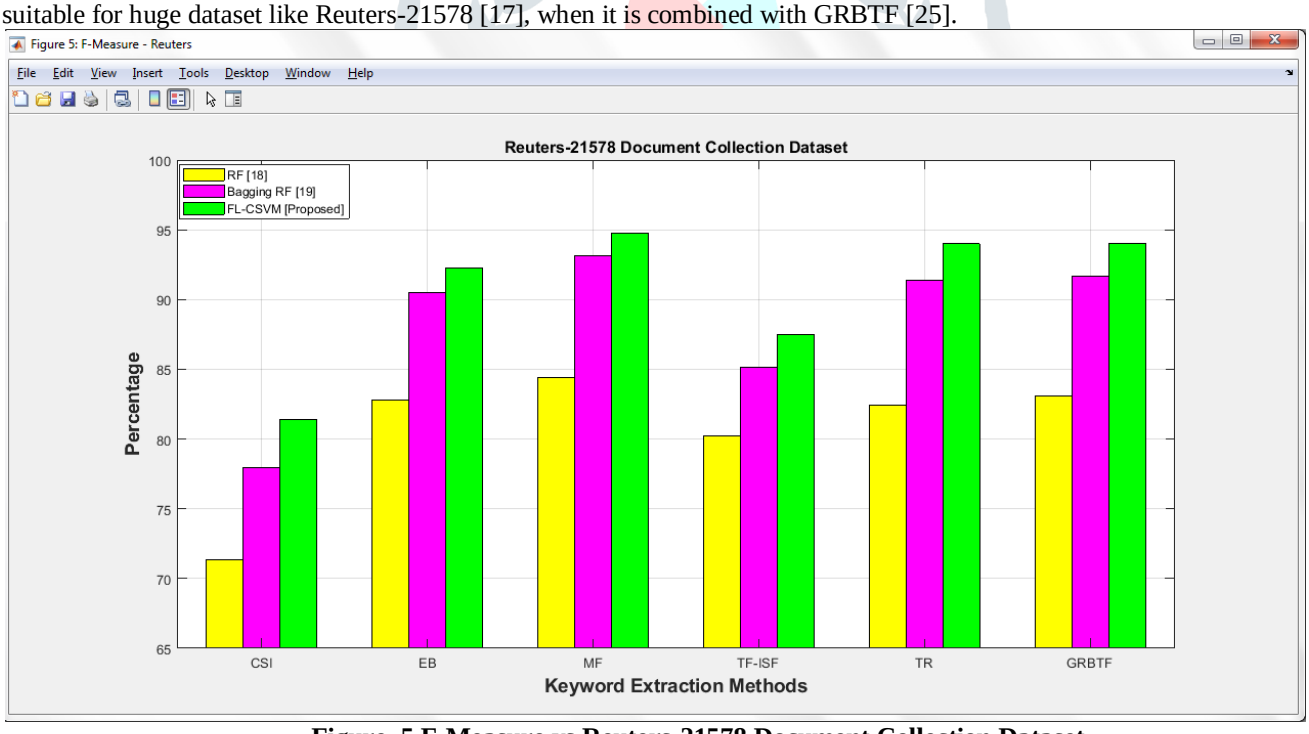


Figure. 5 F-Measure vs Reuters-21578 Document Collection Dataset

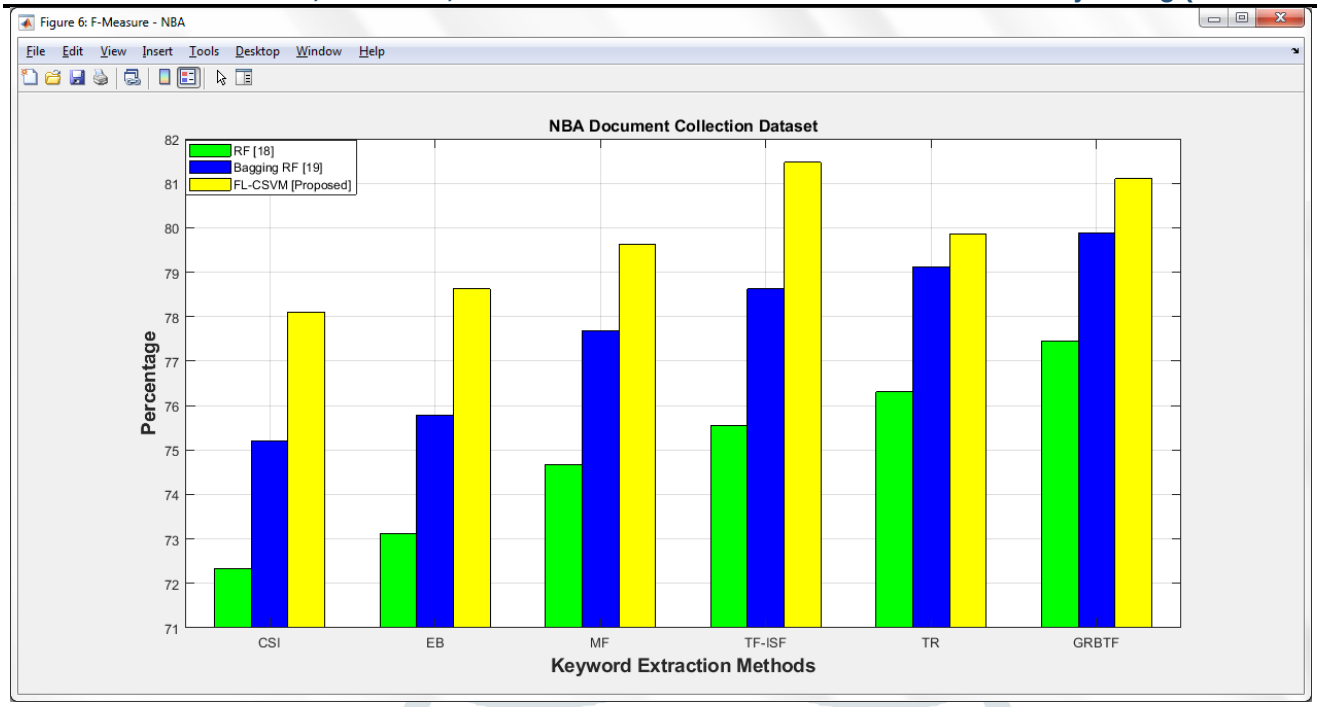


Figure. 6 F-Measure vs NBA Document Collection Dataset

Fig. 6 highlights the F-Measure results of classification algorithms RF [18] and Bagging RF [19] and the proposed classifier FL-CSVM. Fig. 6 clearly illustrate that the proposed classifier FL-CSVM having the better F-Measure result with all keyword extraction methods (CSI [20], EB [21], MF [22], TF-ISF [23], TR [24] and specifically with GRBTF [25]), where it has given the low F-Measure with CSI [20]. When analyzing the results, it is found that the keywords extracted by CSI [20] is not sufficient to classify the documents, but FL-CSVM is able to give highest F-Measure when comparing with RF [18] and Bagging RF [19]. The reason for the best outcome of F-Measure by FL-CSVM is, it does not take all the documents at once processing the classification, instead it sets the threshold value and takes the documents for processing in a batch. The results shows that FL-CSVM is best suitable NBA Input Document Collection Dataset, when it is combined with GRBTF [25].

8. CONCLUSION

FL-CSVM is proposed in the view to perform document classification with increased accuracy. FL-CSVM works with the keywords extracted by different methods. The existing classifiers are available only for small or specific dataset, and in specific it does not have good performance with dataset that are huge in size. FL-CSVM is designed to perform classification with the dataset that are in any size. Classification is performed by dividing the dataset into multiple random groups and it is done to increase the classification accuracy and f-measure. ACM document collection dataset, Reuters-21578 document collection dataset, and NBA Input Document Collection Dataset are used for evaluating the performance of FL-CSVM against existing classifiers. Future direction of this research work can be focused with increasing the classification even more.

References

1. M. Kang, J. Ahn, K. Lee, "Opinion Mining Using Ensemble Text Hidden Markov Models for Text Classification", *Expert Systems with Applications*, Volume 94, 2018, Pages 218-227.
2. N. Sharma, P. Shivakumara, U. Pal, M. Blumenstein, C. L. Tan, "Piece-wise linearity based method for text frame classification in video", *Pattern Recognition*, Volume 48, Issue 3, 2015, Pages 862-881.
3. N. Chi, K. Lin, N. El-Gohary, S. Hsieh, "Evaluating the strength of text classification categories for supporting construction field inspection", Automation in Construction, Volume 64, 2016, Pages 78-88.
4. D. AbuZeina, F. S. Al-Anzi, "Employing fisher discriminant analysis for Arabic text classification", *Computers & Electrical Engineering*, Volume 66, 2018, Pages 474-486.
5. N. Sanchez-Pi, L. Martí, A. C. B. Garcia, "Improving ontology-based text classification: An occupational health and security application", *Journal of Applied Logic*, Volume 17, 2016, Pages 48-58.
6. M. Pavlinek, V. Podgorelec, "Text classification method based on self-training and LDA topic models, *Expert Systems with Applications*, Volume 80, 2017, Pages 83-93.
7. W. Zhang, X. Tang, T. Yoshida, "TESC: An approach to TExt classification using Semi-supervised Clustering", *Knowledge-Based Systems*, Volume 75, 2015, Pages 152-160.
8. Z. E. Rasjid, R. Setiawan, "Performance Comparison and Optimization of Text Document Classification using k-NN and Naïve Bayes Classification Techniques", *Procedia Computer Science*, Volume 116, 2017, Pages 107-112.
9. V. K, D. Gupta, "Text plagiarism classification using syntax based linguistic features*", Expert Systems with Applications*, Volume 88, 2017, Pages 448-464,
10. A. S. Vieira, L. Borrajo, E.L. Iglesias, "Improving the text classification using clustering and a novel HMM to reduce the dimensionality", *Computer Methods and Programs in Biomedicine*, Volume 136, 2016, Pages 119-130.
11. K. Berkling, R. Lavalley, "Automatic orthographic error tagging and classification for German texts", *Computer Speech & Language*, Volume 52, 2018, Pages 56-78.

12. Z. Wu, H. Zhu, G. Li, Z. Cui, H. Huang, J. Li, E. Chen, G. Xu, "An efficient Wikipedia semantic matching approach to text document classification", *Information Sciences*, Volume 393, 2017, Pages 15-28.
13. L. Wang, J. Niu, H. Song, M. Atiquzzaman, Senti, "Related: A cross-domain sentiment classification algorithm for short texts through sentiment related index", *Journal of Network and Computer Applications*, Volume 101, 2018, Pages 111-119.
14. J. Xu, P. Shivakumara, T. Lu, C. L. Tan, S. Uchida, "A new method for multi-oriented graphics-scene-3D text classification in video", *Pattern Recognition*, Volume 49, 2016, Pages 19-42.
15. J. Du, "Automatic text classification algorithm based on Gauss improved convolutional neural network", *Journal of Computational Science*, Volume 21, 2017, Pages 195-200.
16. R. G. Rossi, R. M. Marcacini, S. O. Rezende, Analysis of Domain Independent Statistical Keyword Extraction Methods for Incremental Clustering, *Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence*, Vol. 12, Iss. 1, 2014, Pages 17-37.
17. A. K. Uysal, "An Improved Global Feature Selection Scheme for Text Classification", *Expert Systems with Applications*, Vol. 43, 2016, Pages 82-92.
18. L. Breiman, Random Forests, *Machine Learning*, Vol. 45, Issue. 1, 2001, Pages 5-32
19. A. Onan, S. Korukoglu, H. Bulut, "Ensemble of Keyword Extraction Methods and Classifiers in Text Classification", *Expert Systems with Applications*, Vol. 57, 2016, Pages 232-247.
20. Y. Matsuo, M. Ishizuka, "Keyword Extraction from a Single Document using Word Co-Occurrence Statistical Information", *International Journal on Artificial Intelligence Tools*, Vol. 13, Issue. 1, pages. 157–169, 2004.
21. G. K. Palshikar, "Keyword Extraction from a Single Document Using Centrality Measures", In: Proc. *Second International Conference on Pattern Recognition and Machine Intelligence*, India. *Lecture Notes in Computer Science*, Vol 4815, pages 503-510, 2007.
22. R. G. Rossi, R. M. Maracini, S. O. Rezende, "Analysis of Domain Independent Statistical Keyword Extraction Methods for Incremental Clustering", *Learning and Nonlinear Models*, Vol. 12, Issue. 1, pages 17–37, 2014.
23. L. J. Neto, A . D. Santos, C. A .Kaestner, A. A. Freitas, "Document Clustering and Text Summarization", In: Proc. *4th International Conference on Practical Applications of Knowledge Discovery and Data Mining*, United Kingdom, pp. 41–55, 2000.
24. R. Mihalcea, P. Tarau, "TextRank: Bringing Order into Text", In: Proc. *2004 Conference on Empirical Methods in Natural Language Processing*, Spain, pp. 404–411, 2004.
25. D. Roland, F. Dick, J.L. Elman, Frequency of Basic English Grammatical Structures: A Corpus Analysis, Journal of Memory and Language, Vol 57, Iss. 3, 2007, Pages 348-379.