Journal of High Speed Networks 27 (2021) 183–202
DOI 10.3233/JHS-210659
IOS Press


# Incremental text categorization based on hybrid optimization-based deep belief neural network

V. Srilakshmi [a,*], Dr. K. Anuradha [b] and Dr. C. Shoba Bindu [c]
[a] *Research Scholar, CSE, JNTUA, Anantapur, India*
[b] *Professor, CSE, GRIET, Hyderabad, India*
[c] *Professor, CSE, JNTUA, Anantapur, India*

**Abstract.** One of the effective text categorization methods for learning the large-scale data and the accumulated data is incremental learning. The major challenge in the incremental learning is improving the accuracy as the text document consists of numerous terms. In this research, a incremental text categorization method is developed using the proposed Spider Grasshopper Crow Optimization Algorithm based Deep Belief Neural network (SGrC-based DBN) for providing optimal text categorization results. The proposed text categorization method has four processes, such as are pre-processing, feature extraction, feature selection, text categorization, and incremental learning. Initially, the database is pre-processed and fed into vector space model for the extraction of features. Once the features are extracted, the feature selection is carried out based on mutual information. Then, the text categorization is performed using the proposed SGrC-based DBN method, which is developed by the integration of the spider monkey optimization (SMO) with the Grasshopper Crow Optimization Algorithm (GCOA) algorithm. Finally, the incremental text categorization is performed based on the hybrid weight bounding model that includes the SGrC and Range degree and particularly, the optimal weights of the Range degree model is selected based on SGrC. The experimental result of the proposed text categorization method is performed by considering the data from the Reuter database and 20 Newsgroups database. The comparative analysis of the text categorization method is based on the performance metrics, such as precision, recall and accuracy. The proposed SGrC algorithm obtained a maximum accuracy of 0.9626, maximum precision of 0.9681 and maximum recall of 0.9600, respectively when compared with the existing incremental text categorization methods.

Keywords: Text categorization, deep belief neural network, incremental learning, hybrid optimization, vector space model



## 1. Introduction

The rapid growth of the internet has increased the number of documents available online. Nowadays countless documents are available online, and some of the regular documents include technical reports, newspaper, research articles, and journal papers, etc. A text cloud [10] is a visualization of word frequency in a given text as a weighted list. These text documents are more valuable and useful [23]. Effective retrieval methods are required for the enormous number of documents. One of the active and important areas in information retrieval technology is Text Classification. The documents are categorized into a predefined categories depending on the contents using Text Classification [30]. In the Text Classification, the valuable information are extracted from diverge and numerous online textual documents [9]. TC is a data mining approach that used in the applications, like news monitoring, spam and legitimate email filtering as well as for searching information that are useful on the web. The main aim of the Text Classification is the classification of the document into set of category by extracting the valuable information from unstructured textual resource [2]. The massive information in the text documents made text

*Corresponding author. E-mail: potlurisrilakshmi@gmail.com.

mining a tedious process. The derived linguistic features from the text are used in the process of text mining. Various algorithms are developed for handling the classification of text and to increase the efficiency [22].

Text categorization helps in the detection of fraudulent documents, filtering the spam mails, analyzing the sentiments and for spotting the topics [35]. The text classification is used in the domains, such as email filtering [32], type of the subject or topic [12], SMS spam filtering [8], personality prediction [18], author identification [5], sentiment analysis [20] and web page classification [24]. The text document is categorized based on the features extracted from the documents [21]. Text classification is one of the significant tasks in the supervised Machine Learning (ML) [6,7]. The most commonly used ML techniques for the classification of the text are Naive Bayes (NB) [4], associative classification (AC) [11], K-nearest neighbor (KNN), Random Forest (RF) [34], and support vector machine (SVM) [15]. The intelligence data, which contains a large number of data are enhanced with the Incremental learning algorithms. The algorithms employed for the categorization of the text are Neural networks and Decision trees [19]. For determining the issues from online social media, the Kullback Leibler Distance (KLD) is used by the incremental text classifier [33]. Enhanced incremental learning performance is provided through the text classification using the Naive Bayes. Although the solutions from the categorization are quite easier and effective, the complication lies in the estimation of certain parameters [16].

The vector space is used for designing the documents in the text categorization in which every word is considered as a feature. The values of the features in the vector space model (VSM) are termed as term frequency-inverse document frequency (TF-IDF) or word frequency. The addressing of feature space with large dimensionality is the major issue in text categorization. The classification accuracy is degraded, and the computational time is maximized due to the large set of features. Thus, the dimensionality of the feature space of text is minimized by the effective selection and the extraction of the features. The feature extraction is done by converting or integrating the original to produce new feature set. The minimization of the space dimension is carried out by selecting the most protuberant feature [25]. The feature selection methods are classified into three types, such as embedded, wrapper, and filters. Various filter mechanism developed for the text categorization is Chi-square ($\chi 2$), Information Gain (IG), and Document Frequency (DF) [41]. The term dependencies are captured by employing the N-gram language model that is trained based on the corpus. The Jeffreys-Multi Hypothesis (JMH) divergence is one of the feature selection methods used for the effective categorization of text [36].

The major intention of this research is to develop a technique for incremental text categorization using the proposed classifier. Initially, text categorization is initiated using the pre-processing step, which is applied to the database. Then, the pre-processed data is fed to the VSM for the extraction of the features. Once the features are extracted, the selection of the features are evaluated based on the mutual information. These features altogether represent the feature vector and forms as the input to the text categorization model. The text categorization is performed using the proposed Spider Grasshopper Crow Optimization Algorithm-based Deep Belief Neural network (SGrC-based DBN). The proposed SGrC is developed with the integration of SMO [3] with the GCOA, and the proposed SGrC algorithm trains the DBN. Finally, the incremental text categorization is performed depending on the hybrid weight bounding model.

The main contributions of the research are:

**Proposed SGrC-based DBN**: The proposed SGrC is developed by the integration of the SMO algorithm with the GCOA, which further boosts the optimization searchability and diversity. The DBN is trained for the incremental categorization, where the proposed SGrC algorithm optimally tunes the internal model parameters of DBN.

The organization of the research is as follows: Section 1 introduces the text categorization, Section 2 discusses the text categorization methods, Section 3 explains the Incremental text categorization using proposed SGrC-based DBN, Section 4 discusses the result of the proposed SGrC-based DBN method, and Section 5 concludes the paper.

## 2. Motivation

This section explains the existing incremental text categorization methods along with the advantages and the limitations. Moreover, the challenges faced during the incremental text categorization methods are also discussed.

### 2.1. Literature review

The review of the existing works is explained in this section. Ranjan and Prasad [26] developed a hybrid fuzzy neural network for the classification of text. In this method, the space dimensionality was reduced using the entropy model. The Back Propagation Lion Neural Network was modified with the fuzzy bounding method in which the neuron weights were determined with the Lion Algorithm. The classification was performed using the error estimate without the consideration of old instances. However, the classification efficiency needs to be enhanced. Xu et al. [40] designed ISVM based on Markov resampling (MR-ISVM). This method explained the influence of the sampling methods on the ISVM's learning performance. Although this method had small misclassification rates and low sampling time, it failed to overcome the multiclass classification problem. Sanghani and Kotecha [28] modelled an incremental personalized email spam filter for the classification of text. Initially, the discriminating features were selected by applying the category ratio and frequency difference. Then, the discrimination function was updated dynamically using the incremental learning model. Finally, the features were obtained using the selection Rank Weight-based heuristic function. This method reduced the false-positive error and improved the classification accuracy. However, this method failed to make unique classification decisions for different filters. Park and Kim [25] developed a hierarchical classification for incremental class learning using the adaptive resonance theory-supervised predictive mapping (ARTMAP). The input data that are sequentially added and had different classes were learned incrementally using ARTMAPHC. The class dependency was reflected by adopting prior labels appending process for the hierarchy level. This method did not consider any particular knowledge domain for the classification of new data.

Shan et al. [31] developed Learn, an incremental learning method for the classification of text. The text features were extracted for predicting the text category using the neural network model. The reinforcement learning module was used for filtering the predictions. Although this method provided text classification on different datasets, it failed to use RL algorithms for the implementation of the Teacher model. Wohlwend et al. [39] designed a dynamic classification of text using metric learning. This method prevented the overlapping of label sets by combining the low-resource training data. However, this method failed to explore the Wasserstein distance and label entailment. Srivastava et al. [34] designed a machine learning model for the classification of text. The performance of the classifier was improved using the different aspects, such as term frequency-inverse document frequency (TF-IDF), latent semantic indexing, and bag of words (BOW) for the selection of features. Although the classifier interface along with the feature extraction was easy to adapt, it did not had large data types. Jiang et al. [14] developed a hybrid model that depends on softmax regression for the classification of text. The text classification was done through the softmax regression, whereas the sparse matrix and high dimensional problems were solved using DBN. This method avoided the optimization in the indirect route

### 2.2. Challenges

The challenges of the research are:

- In the study by Srivastava et al. [34], the method did not intend to justify with datasets. The challenge lies in the understanding the behaviour of machine learning using various types of the classifier.
- The challenge in the MR-ISVM method is solving the multiclass classification problems, concept drift besides studying the learning performance of the ISVM and MRISVM for the regression based on Markov resampling and nonlinear prediction models, respectively [40].
- Although the incremental learning model reduced the false-positive error and improved the classification accuracy, the main challenge lies in solving the multi-class classification problems [28].
- The challenge in the Learn method lies in forming ensemble classifiers that depend on the dataset distribution for the Student models and using specific RL algorithms for the implementation of the Teacher Model [31].
- The Incremental text categorization-based GCOA method achieved high efficiency and accuracy, but the challenge lies in the determination of relevant information from a large datasets due to the diversity of the conversational language.

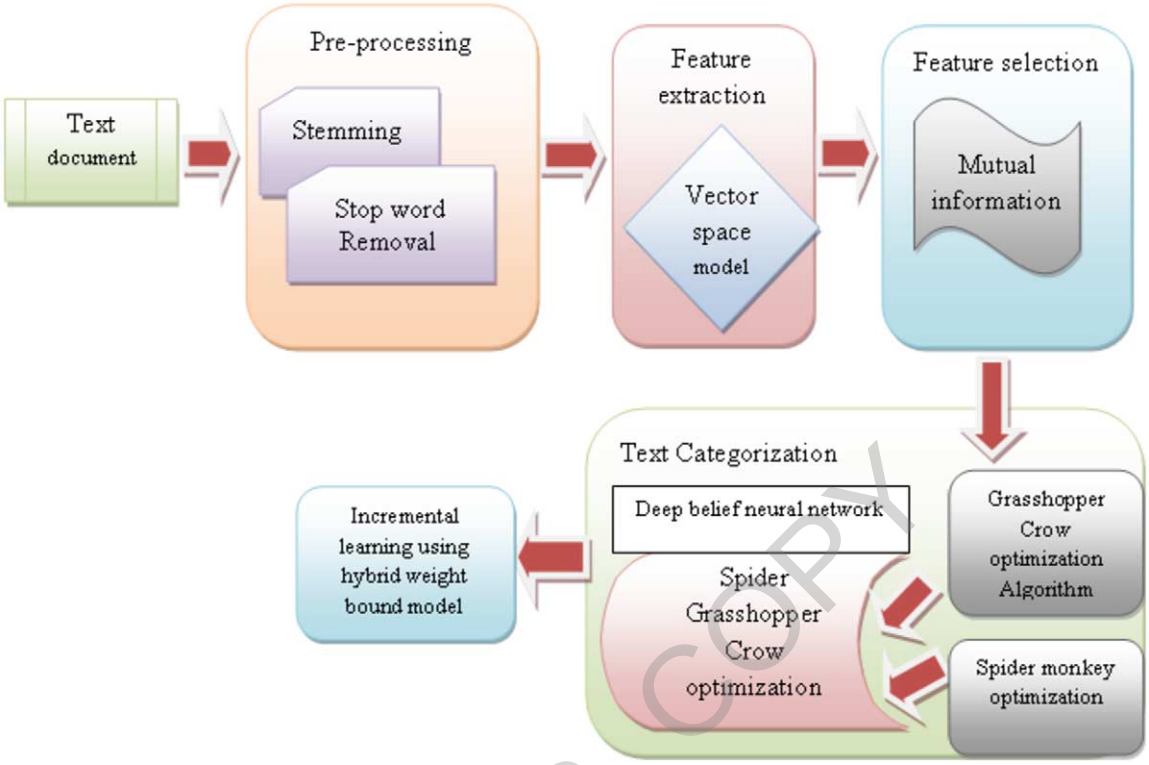


Fig. 1. Block diagram of the text categorization using proposed SGrC based DBN.

## 3. Text incremental categorization using proposed SGrC-based deep belief neural networks

This section describes the text categorization method using the proposed SGrC-based DBN. Initially, the redundant and inconsistent words are eliminated from the documents in the pre-processing phase using the stop word removal and stemming process. The pre-processing process is followed by the feature extraction process in which the VSM is used for extracting the TF-IDF. Then, the best features are selected from the extracted features for performing text categorization. The text categorization is done using the proposed SGrC algorithm, which is the integration of the SMO algorithm [3] into the GCOA. The proposed SGrC algorithm trains the DBN for the categorization of the text. Finally, the incremental text categorization is done depending on the hybrid weight bounding model. The SGrC algorithm determines the optimal weights for the Range degree model. Figure 1 shows the block diagram of the proposed text categorization method using SGrC based DBN.

Let us assume a document $B$ with different attributes as,

$$B = \{B_{c,d}\}; \quad (1 \leqslant c \leqslant C)(1 \leqslant d \leqslant D) \tag{1}$$

where, the data points and the total number of attributes are represented as, $C$ and $D$. In the database $B$, the document with the $d^{th}$ attribute in $c^{th}$ data is denoted as, $B_{c,d}$.

### 3.1. Pre-processing

The redundant words in the text documents are removed by the pre-processing process. There are two main pre-processing steps, such as Stop word removal and Stemming. The pre-processing phase is done in the input

document for smoother processing. The text documents that contain the redundant phrases and words are removed as the text documents with larger sizes are affected by the text categorization process. Thus, it is important to remove the inconsistent and redundant words through pre-processing.

*i) Stop word removal.* The words, such as preposition, articles, or pronouns used in the sentence are known as stop words. Before the data processing, the stop words are filtered out. In this step, the noise in the data is reduced by eliminating the non-information behavior words. The removal of the stop word enables faster processing and avoids the large-space accumulation for effective results. In this method, the stop words, like nouns and verbs are removed from the document.

*ii) Stemming.* The words are transformed into their stems using the stemming process. The word that has the same concept is utilized in large documents. The technique of reducing the word to its root word is defined as stemming. For example, the word 'agree' is related to the words, such as agree, disagree and agreeing. The stemming does not require the suffix list, and in addition, it is easier to use and compact. The words that are non-meaningful from the roots of the language are eliminated in the stemming method.

### *3.2. Extraction of features using vector space model for text categorization*

This section explains the features extracted from the input document. The complexity for analyzing the document is reduced if the documents are represented in terms of feature set. Thus, for the effective document verification, highly relevant features are extracted to obtain the feature vector. In this research, significant feature, such as the TF-IDF feature is extracted from the documents through the vector space model (VSM) for better categorization of the text. The text documents are represented as vectors using the VSM [17]. The VSM is the algebraic model applied for various applications, like filtering the information, retrieving the information, relevant ranking, and indexing. Thus, the TF-IDF feature extracted from the documents with the help of VSM is explained below:

#### *3.2.1. Computation of the occurrence of words by the extraction of the TF-IDF features*

The occurrence of the each and every word in the document is determined by the TF computation, whereas the words that rarely occur in the document is determined using IDF. The equation formulated from the TF-IDF is given as,

$$QR(STU) = Q(ST) \times R(SU) \tag{2}$$

where, the rate of occurrence of words in the document is termed as, $Q(ST)$, the frequency of computation of the important words that occurs in the document rarely is termed as, $R(SU)$, the total number of words and the total number of documents is represented as, $U$ and $S$, where $(1 \leqslant T \leqslant U)$. The IDF equation is given as,

$$R(SU) = \log \frac{|U|}{1 + |\{T \in U : S \in T\}|} \tag{3}$$

### *3.3. Feature selection for categorizing texts*

The information gain is applied for selecting the significant features from the extracted features. The feature selection process enhanced the classifier performance along with the reduction in the search space dimension. It also minimized the size of the index.

#### *3.3.1. Selection of relevant features using information gain*

One of the popular methods employed in the selection of the relevant features for the categorization of text in the text document data is the information gain [38]. The information theory is the idea behind the information gain. The information gain for the term $\tilde{k}$ is given as,

$$I_G(\overleftrightarrow{k}) = -\sum_{y=1}^{|\bar{C}|} P(x_i) \log P(x_i) + P(\hat{k}) \sum_{y=1}^{|\bar{C}|} P(x_i/\hat{k}) \log P(x_i/\hat{k}) + P(\bar{k}) \times \sum_{y=1}^{|\bar{C}|} P(x_i/\bar{k}) \log P(x_i/\bar{k}) \tag{4}$$

where, probability of the $i$th category $x_i$ is denoted as, $P(x_i)$. The probability of the appearance and the absence of the term $\hat{k}$ in the document is given as, $P(\hat{k})$ and $P(\bar{k})$, $i$th category conditional probability for the appearance of the term $\hat{k}$ in the document is given as, $P(x_i/\hat{k})$ and $i$th category conditional probability for the absence of the term $\tilde{k}$ in the document is given as, $P(x_i/\bar{k})$. Depending on the number of documents, the feature vector is represented as,

$$\hat{r} = [\hat{B} \times \hat{j}] \tag{5}$$

where, the total number of documents and the unique word count are represented as, $\hat{B}$ and $\hat{j}$.

### 3.4. Proposed SGrC based DBN for the text categorization

This section illustrates the text categorization process using the proposed SGrC-based DBN. The equivalent class of the query is determined, and the weight of the DBN is updated by the proposed SGrC-based DBN during the arrival of the new query. The optimal weights are obtained by training the DBN [13] using the proposed SGrC method, and the DBN provides accurate results. The proposed SGrC is developed by the integration of the SMO algorithm and the GCOA. The GCOA provides an effective categorization of text as it is obtained by the combination of the GOA and CSA algorithm. The GOA solves the optimization problems by obtaining the best solution besides balancing the exploration and exploitation. The drawback of the GOA is the poor convergence rate. The CSA algorithm provides a better convergence rate. Thus, the GCOA resolves the issues of both the GOA and CSA by integrating them. On the other hand, the SMO algorithm depends on the social behavior of the spider monkeys. The distance of the food source is estimated by updating the food source, and the feature score is determined by updating the location of the local and global leader group members. The proposed SGrC algorithm provides accurate results by tuning the weights of DBN.

#### 3.4.1. Architecture of DBN classifier

The DBN consists of Restricted Boltzmann Machines (RBMs) layers and Multilayer Perceptrons (MLPs) layers that are connected with the weights [34]. The RBMs consist of visible layer and hidden layer, whereas the MLPs consist of the input layer, hidden layer and output layer. Figure 2 shows the architecture of the DBN classifier for the determination of incremental text categorization.

The features from the database is given as the input to the RBM_1 visible layer which is represented as,

$$u^1 = \left\{u_1^1, u_2^1, \ldots, u_k^1, \ldots, u_m^1\right\}; \quad 1 \leqslant k \leqslant m \tag{6}$$

where, the $m$th visible neuron in the RBM_1 layer is represented as, $r_m^1$. The RBM_1 hidden layer is represented as,

$$o^1 = \left\{o_1^1, o_2^1, \ldots, o_z^1, \ldots, o_n^1\right\}; \quad 1 \leqslant z \leqslant n \tag{7}$$

where, the $z^{th}$ hidden neuron in the RBM_1 layer is represented as, $o_z^1$ and the total hidden neurons are represented as, $n$. The weight in the RBM_1 layer is represented as,

$$\chi^1 = \left\{\chi_{k,z}^1\right\}; \quad 1 \leqslant k \leqslant 12; 1 \leqslant z \leqslant n \tag{8}$$

where, the weight within the $z^{th}$ hidden neuron and $k$th visible neuron is represented as,
$\chi_{k,z}^1$. The output of the RBM_1 is represented as,

$$o^1 = \left\{o_z^1\right\}; \quad 1 \leqslant z \leqslant n \tag{9}$$

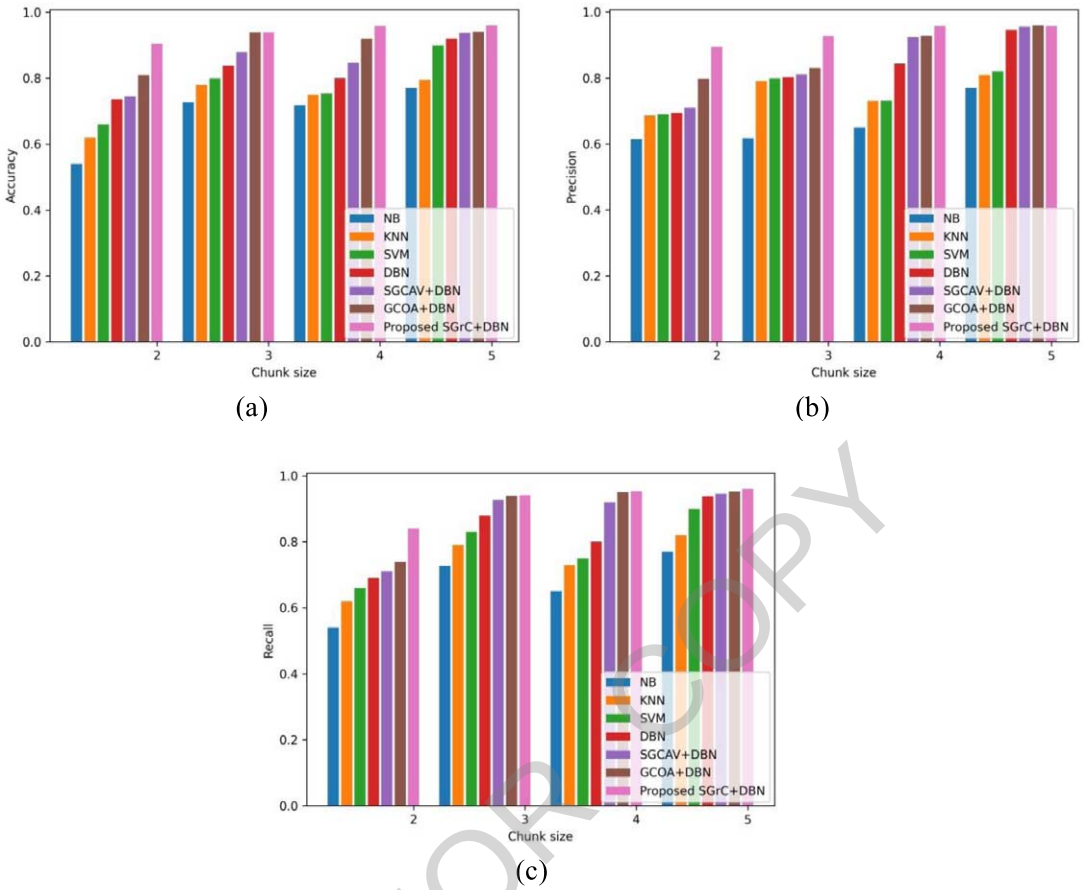


Fig. 2. Architecture of DBN classifier.

where, $o_z^1 = \beta[a_z^1 + \sum_k u_k^1 \chi_{k,z}^1]$ in which $\beta$ denotes the activation function and the $o_z^1$ is the RBM_1's output vector, the bias of the $z^{th}$ hidden neuron is represented as, $a_z^1$. The input to the RBM_2 is the output from the RBM_1. In the RBM_2, the visible layer is given as,

$$u^2 = \{u_1^2, u_2^2, \ldots, u_n^2\} = \{o_z^1\}; \quad 1 \leqslant z \leqslant n \tag{10}$$

In the RBM_2, the hidden layers and the weight vectors are represented as,

$$o^2 = \{o_1^2, o_2^2, \ldots, o_z^2, \ldots, o_n^2\}; \quad 1 \leqslant z \leqslant n \tag{11}$$

$$\chi^2 = \{\chi_{zz}^2\}; \quad 1 \leqslant z \leqslant n \tag{12}$$

where, the weight between the $z^{th}$ hidden neuron and $z^{th}$ visible neuron in the RBM_2 is represented as, $\chi_{zz}^2$. The hidden layer output from the RBM_2 is given as,

$$o^2 = \{o_z^2\}; \quad 1 \leqslant z \leqslant n \tag{13}$$

where, $o_z^2$ is the output of the $z^{th}$ hidden neuron in RBM_2, which is given as, $o_z^2 = \beta[a_z^2 + \sum_k u_k^2 \chi_{zz}^2] \forall u_k^2 = o_z^1$. The MLP layer's input is represented as,

$$h = \{h_1, h_2, \ldots, h_z, \ldots, h_n\} = \left\{o_z^2\right\}; \quad 1 \leqslant z \leqslant n \tag{14}$$

where, the input layer neurons are represented as, $n$. The above equation is given as the input to the MLP's hidden layer. The output of the MLP's hidden layer is represented as,

$$w = \{w_1, w_2, \ldots, w_I, \ldots, w_U\}; \quad 1 \leqslant I \leqslant V \tag{15}$$

where, the bias of the $o^{th}$ hidden neuron is given as, $n_o$ and the number of the hidden neurons in the MLP layer is given as, $M$. The output layer of the MLP is given as,

$$Z = \{Z_1, Z_2, \ldots, Z_e, \ldots, Z_f\}; \quad 1 \leqslant e \leqslant f \tag{16}$$

where, the neurons in the output layer are represented as, $f$. The two-weight vector in the MLP layer is given as,

$$\chi' = \left\{\chi'_{ze}\right\}; \quad 1 \leqslant z \leqslant n; 1 \leqslant e \leqslant V \tag{17}$$

$$\chi'' = \left\{\chi''_{Ie}\right\}; \quad 1 \leqslant I \leqslant M; 1 \leqslant e \leqslant f \tag{18}$$

where, the weight within the hidden layer and the output layer is denoted as, $\chi''$, the weight within the $e^{th}$ hidden neuron and the $z^{th}$ input neuron is represented as, $\chi'_{ze}$. The weight between the input layer and the hidden layer is given as, $\chi'$. The output from the MLP's hidden layer is given as,

$$w_o = \left[\sum_{z=1}^{n} \chi''_{zI} * X_z\right] O_e \quad \forall X_z = o_z^2 \tag{19}$$

where, the hidden neuron's bias is given as, $O_e$. The final output vector of the DBN is given as,

$$Z_e = \sum_{I=1}^{V} \beta''_{eI} * w_e \tag{20}$$

The output vector is obtained from the hidden layer's output and the weight, $\omega''$. In the above equation, the weight between the $o^{th}$ output neuron and the $O^{th}$ hidden neuron is represented as, $\omega'''_{oO}$, the hidden layer's output is represented as, $n_o$.

*3.4.2. Training DBN using SGrC algorithm*

This section discusses the proposed SGrC algorithm for training the DBN. The SGrC algorithm is developed by integrating the SMO algorithm in the GCOA algorithm. The optimal text categorization results are provided by the proposed SGrC-based DBN method. Finally, the incremental text categorization is developed based on the hybrid weight bounding model that includes the SGrC and Range degree and particularly, SGrC aims at the selection of the optimal weights for the Range degree model. The algorithmic description of the proposed SGrC algorithm is as follows:

*(a) Initialization.* Initially, the weights of the DBN are randomly initialized as,

$$W = \{W_1, W_2, \ldots, W_l, \ldots W_\beta\}; \quad 1 < l \leqslant \beta \tag{21}$$

where, the total weights are represented as, $\beta$.

*(b) Estimation of error.* The output is obtained by applying the selected features, $K$ and weight, $W$ to the DBN. The sum of the squares of the network's current output is the output error. The network is trained using the output label, which is represented as,

$$M^{\tau+1} = \frac{1}{A}\sum_{v=1}^{A}\left[F_v^{\tau} - L_v^{\tau}\right] \tag{22}$$

where, total data samples, predicted output and the estimated output at current iteration is represented as, $L_v^{\tau}$ and $F_v^{\tau}$.

*(c) Weight update using proposed SGrC.* The weight is updated based on the proposed SGrC algorithm, and the equation is derived based on the weights obtained from integrating the GCOA algorithm and the SMO algorithm. According to the GCOA algorithm derived in the previous work, the weight obtained is denoted as,

$$W_i(\tau+1) = \frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau) - 1} c\left(\sum_{\substack{j=1 \\ j\neq i}}^{N} c\frac{E_g - F_g}{2} s\left(\left|b_j^g - b_i^g\right|\right)\frac{b_j - b_i}{g_{i,j}}\right) + \frac{W_i(\tau)}{r_i \times J_i(\tau)}\left[r_i \times J_i(\tau) - 1\right] \tag{23}$$

where, the upper and the lower bound in the $g^{th}$ dimension is given as, $E_g$ and $F_g$, the decreasing coefficient and the social forces are denoted as $c$ and $s$, In the $g^{th}$ dimension, the position of the $i$th and $j$th grasshopper is denoted as, $b_i^g$ and $b_j^g$. The distance between $i$th grasshopper and $j$th grasshopper is given as, $g_{i,j}$. At the current iteration $\tau$, position of the crow and flight length of $i$th dimension is given as, $W_i(\tau)$ and $J_i(\tau)$. According to the SMO algorithm, the weight obtained is given as,

$$W_i(\tau+1) = W_i(\tau) + P(0,1)\big(H - W_i(\tau)\big) + P(-1,1)\big(W_j(\tau) - W_i(\tau)\big) \tag{24}$$

$$W_i(\tau+1) = W_i(\tau) + P(0,1)H - P(0,1)W_i(\tau) + P(-1,1)W_j(\tau) - P(-1,1)W_i(\tau) \tag{25}$$

$$W_i(\tau+1) = W_i(\tau) + \big(1 - P(0,1) - P(-1,1)\big) + P(0,1)H + P(-1,1)W_j(\tau) \tag{26}$$

$$W_i(\tau) = \frac{W_i(\tau+1) - P(0,1)H - P(-1,1)W_j(\tau)}{1 - P(0,1) - P(-1,1)} \tag{27}$$

Substituting in Eq. (23), we get

$$\begin{aligned} W_i(\tau+1) = {} & \frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau) - 1} c\left(\sum_{\substack{j=1 \\ j\neq i}}^{N} c\frac{E_g - F_g}{2} s\left(\left|b_j^g - b_i^g\right|\right)\frac{b_j - b_i}{g_{i,j}}\right) \\ & + \frac{W_i(\tau+1) - P(0,1)H - P(-1,1)W_j(\tau)}{1 - P(0,1) - P(-1,1)} \frac{r_i \times J_i(\tau) - 1}{r_i \times J_i(\tau)} \end{aligned} \tag{28}$$

$$\begin{aligned} W_i(\tau+1) = {} & \frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau) - 1} c\left(\sum_{\substack{j=1 \\ j\neq i}}^{N} c\frac{E_g - F_g}{2} s\left(\left|b_j^g - b_i^g\right|\right)\frac{b_j - b_i}{g_{i,j}}\right) \\ & + \frac{W_i(\tau+1)}{1 - P(0,1) - P(-1,1)} \frac{r_i \times J_i(\tau) - 1}{r_i \times J_i(\tau)} - \frac{P(0,1)H - P(-1,1)W_j(\tau)}{1 - P(0,1) - P(-1,1)} \frac{r_i \times J_i(\tau) - 1}{r_i \times J_i(\tau)} \end{aligned} \tag{29}$$

$$W_i(\tau+1) = \frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau) - 1} \frac{W_i(\tau+1)}{1 - P(0,1) - P(-1,1)} \frac{r_i \times J_i(\tau) - 1}{r_i \times J_i(\tau)}$$

$$+\frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau)-1} c\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\frac{E_g-F_g}{2} s\left(\left|b_j^g-b_i^g\right|\right)\frac{b_j-b_i}{g_{i,j}}\right)$$

$$-\frac{P(0,1)H-P(-1,1)W_j(\tau)}{1-P(0,1)-P(-1,1)}\frac{r_i \times J_i(\tau)-1}{r_i \times J_i(\tau)} \tag{30}$$

$$W_i(\tau+1)-\frac{W_i(\tau+1)}{1-P(0,1)-P(-1,1)}$$

$$=\frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau)-1} c\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\frac{E_g-F_g}{2} s\left(\left|b_j^g-b_i^g\right|\right)\frac{b_j-b_i}{g_{i,j}}\right)$$

$$-\frac{P(0,1)H-P(-1,1)W_j(\tau)}{1-P(0,1)-P(-1,1)}\frac{r_i \times J_i(\tau)-1}{r_i \times J_i(\tau)} \tag{31}$$

$$W_i(\tau+1)\left[1-\frac{W_i(\tau+1)}{1-P(0,1)-P(-1,1)}\right]$$

$$=\frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau)-1} c\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\frac{E_g-F_g}{2} s\left(\left|b_j^g-b_i^g\right|\right)\frac{b_j-b_i}{g_{i,j}}\right)$$

$$-\frac{P(0,1)H-P(-1,1)W_j(\tau)}{1-P(0,1)-P(-1,1)}\frac{r_i \times J_i(\tau)-1}{r_i \times J_i(\tau)} \tag{32}$$

$$W_i(\tau+1)\left[\frac{1-P(0,1)-P(-1,1)-1}{1-P(0,1)-P(-1,1)}\right]$$

$$=\frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau)-1} c\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\frac{E_g-F_g}{2} s\left(\left|b_j^g-b_i^g\right|\right)\frac{b_j-b_i}{g_{i,j}}\right)$$

$$-\frac{P(0,1)H-P(-1,1)W_j(\tau)}{1-P(0,1)-P(-1,1)}\frac{r_i \times J_i(\tau)-1}{r_i \times J_i(\tau)} \tag{33}$$

$$W_i(\tau+1)=-\left[\frac{1-P(0,1)-P(-1,1)}{P(0,1)+P(-1,1)}\right]$$

$$\times\left\{\frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau)-1} c\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\frac{E_g-F_g}{2} s\left(\left|b_j^g-b_i^g\right|\right)\frac{b_j-b_i}{g_{i,j}}\right)\right.$$

$$\left.-\frac{P(0,1)H-P(-1,1)W_j(\tau)}{1-P(0,1)-P(-1,1)}\frac{r_i \times J_i(\tau)-1}{r_i \times J_i(\tau)}\right\} \tag{34}$$

$$W_i(\tau+1)=\frac{P(0,1)+P(-1,1)-1}{P(0,1)+P(-1,1)}$$

**Algorithm 1** Pseudo code of the proposed SGrC based DBN

Input: $K$
Output: Best optimal solution
Initialize the number of iterations, population, $c_{\max} = 1$, $c_{\min} = 0.00001$
$\hat{S}$ = best search agent
while ($\hat{g} < L$) $\hat{g}$-attractive length scale
  Update $c$ using the Eq. (36)
    for every search agent
      Update the weight using the Eq. (35)
    end for
    //If the optimal solution is obtained
    Update $\hat{S}$
      $\hat{g} = \hat{g} + 1$
end while
Return $\hat{S}$
Termination

$$\times \left\{ \frac{r_i \times J_i(\tau)}{r_i \times J_i(\tau) - 1} c \left( \sum_{\substack{j=1 \\ j \neq i}}^{N} c \frac{E_g - F_g}{2} s\left(\left|b_j^g - b_i^g\right|\right) \frac{b_j - b_i}{g_{i,j}} \right) \right.$$
$$\left. - \frac{P(0,1)H + P(-1,1)W_j(\tau)}{1 - P(0,1) - P(-1,1)} \frac{r_i \times J_i(\tau) - 1}{r_i \times J_i(\tau)} \right\} \tag{35}$$

where, $r_i$ lies in the range of [0, 1].

$$c = c_{\max} - \tau \left( \frac{c_{\max} - c_{\min}}{L} \right) \tag{36}$$

where, $\tau$ and $L$ are the current and the maximum iteration. $c_{\max} = 1$ and $c_{\min} = 0.00001$. The weight that corresponds to the minimum error value is used for training the DBN as the minimum error describes the best weight.

*(d) Determination of feasible weights.* The weights obtained using the proposed SGrC algorithm are updated as the feasible weight for training the DBN.

*(e) Termination.* Until the maximum iteration limit, the optimal weights are derived in an iterative manner. Algorithm 1 depicts the pseudo code of proposed SGrC based DBN.

### *3.5. Hybrid weight bound for the incremental learning*

The idea of incremental learning using the hybrid weight bound model keeps away the concept drift and paves the way for handling the new incoming data. Here, error plays a dominant role, if the error corresponding to the classified output is large when compared with the target, so that if drift occurs, the classifier model is updated. In the incremental learning algorithm, the error $M^{\tau+1}$ is computed followed by the updation of the weight, whenever a new instance $Y^{\tau+1}$ is added in the network. For the previous instance, if the evaluated error is higher than the computed error then, the weight is generated using the Eq. (35). If the error evaluated is lower than the computed error then, the weight is generated using the hybrid weight model. The hybrid weight model comprises of SGrC and Range degree model. The proposed SGrC chooses the suitable weights for the Range degree model. The weight is

updated based on the difference between the range degree and the stored weights. The equation for evaluating the weight is determined by the below equation as,

$$W(\tau + 1) = W(\tau) \pm G \tag{37}$$

where, the range degree and the stored weights is denoted as, $G$ and $W(\tau)$. The range degree [29] is given as,

$$G = \sqrt{\frac{\upsilon(\log \frac{2\ell}{\upsilon} + 1) - \log(\frac{\mu}{4})}{\ell}} \tag{38}$$

where, the training samples is denoted as, $\ell$, the Vapnik- Chervonenk is (VC) dimension for the function set is represented as, $\upsilon$. The capacity of the function set implemented using the learning machine is the VC dimension. The random number, $\mu$ lies in the range of [0,1].

## 4. Results and discussion

This section describes the result of the proposed SGrC + DBN along with the existing text classification methods based on accuracy, precision, and recall.

### *4.1. Experimental setup*

The proposed SGrC + DBN method is evaluated in the PC with 2 GB RAM, Window 10 OS, Intel i-3 core processor using JAVA software.

### *4.2. Database description*

The dataset used in the proposed SGrC + DBN method for the incremental categorization of text is Reuter database and 20 Newsgroups database.

#### *4.2.1. Newsgroups database*

The 20 Newsgroups data set [1] is developed by accumulating the documents from 20,000 newsgroups that are divided across 20 different newsgroups. The newsgroups represent different topics.

#### *4.2.2. Reuter database*

The Reuters-21578 Text Categorization Collection Data Set [27] consists of documents from the Reuters newswires in 1987 and it is donated by David D. Lewis. The documents are indexed depending on the categories of text. The dataset has five attributes with 21578 number of instances.

### *4.3. Evaluation metrics*

The evaluation metrics used in the proposed SGrC + DBN method are precision, recall, and accuracy.

#### *4.3.1. Precision*

The precision is the nearness of the measurements to the particular value in the text categorization method, and it is given as,

$$\text{Precision} = \frac{t_p}{t_p + f_p} \tag{39}$$

where, the true positive is denoted as, $t_p$ and false positive is denoted as, $f_p$.

*4.3.2. Recall*

The recall is defined by the total number of actual positives in the proposed SGrC + DBN method, and it is represented as,

$$\text{Recall} = \frac{t_p}{t_p + f_n} \tag{40}$$

where, $f_n$ be the false negative.

*4.3.3. Accuracy*

The accuracy is the measure of the closeness in the text categorization method and is represented as,

$$\text{Accuracy} = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \tag{41}$$

*4.4. Comparative analysis*

The proposed SGrC + DBN method is compared with the existing methods based on performance metrics, such as accuracy, precision, and recall using the Reuter database and 20 Newsgroups database.

*4.5. Competing methods*

The competing methods used for the analysis of the proposed SGrC + DBN are NB [29], KNN [37], SVM [15], and DBN [13], SGCAV + DBN, GCOA + DBN.

*4.5.1. Comparative analysis using 20 newsgroup database*

*(a) For feature size* = 100. Figure 3 shows the comparative analysis of the text classification methods for feature size = 100 based on accuracy, precision, and recall. Figure 3(a) depicts the analysis of the text classification methods based on accuracy. For chunk size = 2, the accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5400, 0.6200, 0.6600, 0.7354, 0.7441, 0.8103 and 0.9043, respectively. The accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.7708, 0.7948, 0.9000, 0.9200, 0.9375, 0.9408 and 0.9600, respectively.

Figure 3(b) shows the analysis of the text classification methods based on precision. The precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 2 is 0.6146, 0.6863, 0.6900, 0.6944, 0.7100, 0.7977 and 0.8950, respectively. For chunk size = 5, the precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.7700, 0.8095, 0.8200, 0.9451, 0.9555, 0.9594 and 0.9581, respectively

Figure 3(c) shows the analysis of the text classification methods based on recall. For chunk size = 2, the recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5400, 0.6200, 0.6600, 0.6900, 0.7100, 0.7394 and 0.8402, respectively. The recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.7700, 0.8200, 0.9000, 0.9375, 0.9450, 0.9521 and 0.9600, respectively.

*(b) For feature size* = 200. Figure 4 shows the comparative analysis of the text classification methods for feature size = 200 based on accuracy, precision, and recall. Figure 4(a) depicts the analysis of the text classification methods based on accuracy. For chunk size = 2, the accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.7000, 0.7200, 0.7574, 0.7752, 0.8446, 0.8800 and 0.8873, respectively. The accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.7618, 0.7798, 0.9375, 0.9416, 0.9538, 0.9600 and 0.9626, respectively.

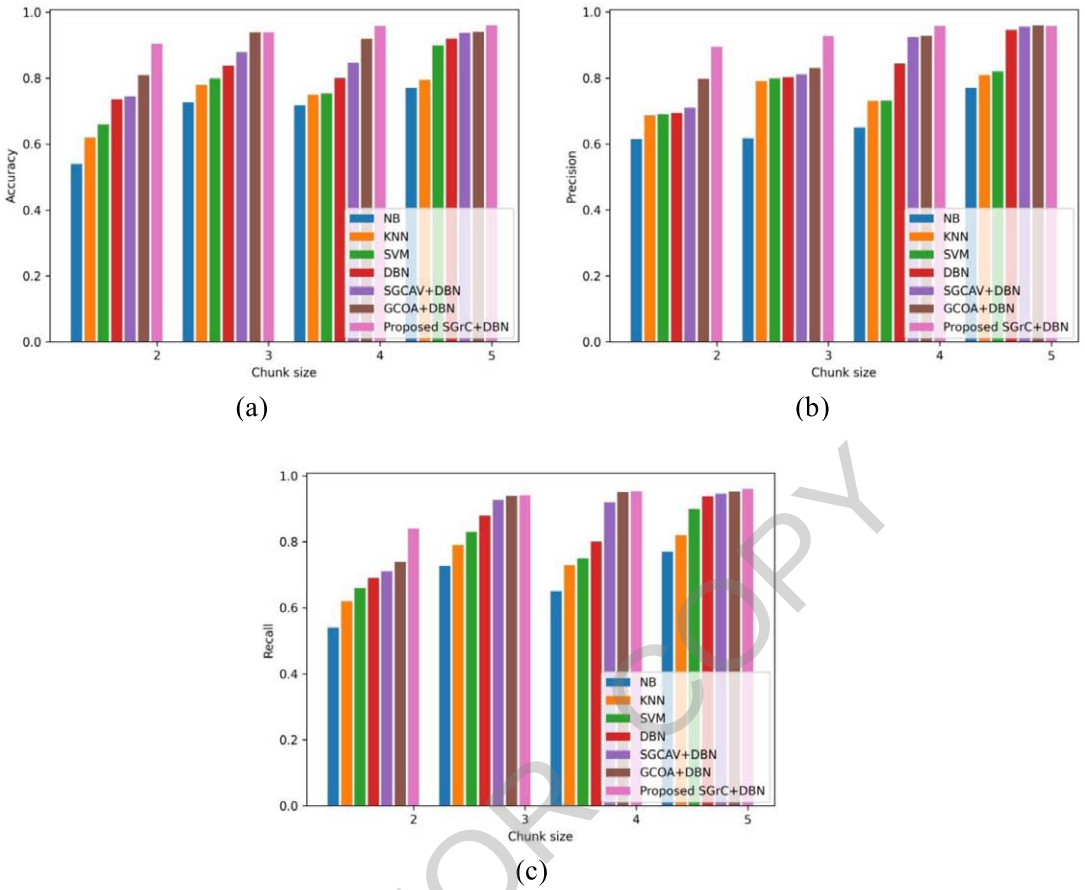


Fig. 3. Analysis of methods for feature size = 100 using (a) Accuracy (b) Precision (c) Recall.

Figure 4(b) depicts the analysis of the text classification methods based on precision. The precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 2 is 0.7400, 0.7647, 0.7746, 0.7800, 0.8592, 0.9391 and 0.9546, respectively. For chunk size = 5, the precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.7500, 0.7900, 0.9048, 0.9427, 0.9539, 0.9596 and 0.9681, respectively

Figure 4(c) depicts the analysis of the text classification methods based on recall. For chunk size = 2, the recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.7000, 0.7200, 0.7400, 0.7800, 0.8014, 0.8800 and 0.9434, respectively. The recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.7500, 0.7900, 0.9375, 0.9480, 0.9585, 0.9590 and 0.9600, respectively.

*4.5.2. Comparative analysis using reuter database*

*(a) For feature size* = 100. Figure 5 shows the comparative analysis of the text classification methods for feature size = 100 based on accuracy, precision, and recall. Figure 5(a) depicts the analysis of the text classification methods based on accuracy. For chunk size = 2, the accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5409, 0.6364, 0.6364, 0.9457, 0.9533, 0.9550 and 0.9579, respectively. The accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.5909, 0.5909, 0.6364, 0.9432, 0.9436, 0.9500 and 0.9592, respectively.

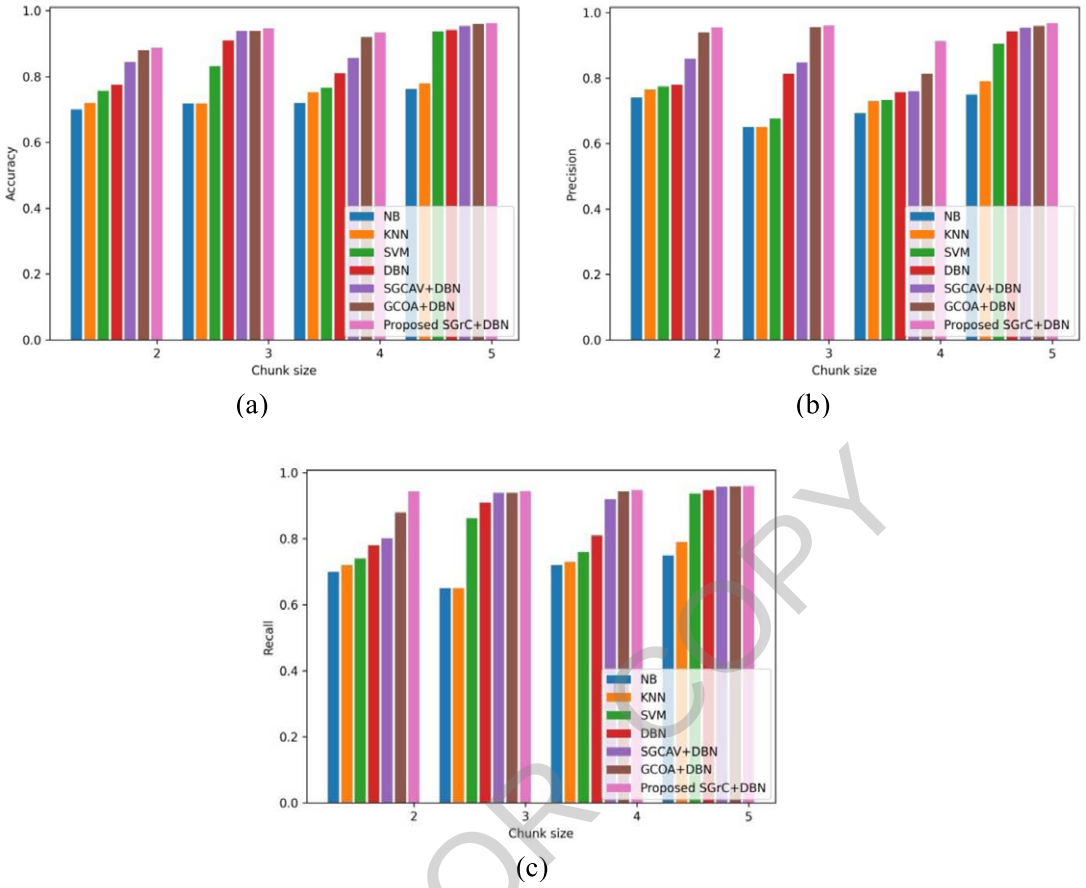


Fig. 4. Analysis of methods for feature size = 200 using (a) Accuracy (b) Precision (c) Recall.

Figure 5(b) depicts the analysis of the text classification methods based on precision. The precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 2 is 0.4005, 0.5954, 0.6964, 0.6978, 0.7273, 0.7636 and 0.7662, respectively. For chunk size = 5, the precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.6050, 0.6087, 0.7178, 0.7428, 0.7455, 0.7727 and 0.8527, respectively

Figure 5(c) depicts the analysis of the text classification methods based on recall. For chunk size = 2, the recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5409, 0.6364, 0.6364, 0.7273, 0.7636, 0.9471 and 0.9537, respectively. The recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.5909, 0.5909, 0.6364, 0.7455, 0.7727, 0.9468 and 0.9538, respectively.

*(b) For feature size* = 200. Figure 6 shows the comparative analysis of the text classification methods for feature size = 100 based on accuracy, precision, and recall. Figure 6(a) depicts the analysis of the text classification methods based on accuracy. For chunk size = 2, the accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5727, 0.6045, 0.6682, 0.9458, 0.9471, 0.9549 and 0.9580, respectively. The accuracy obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.6364, 0.6818, 0.6818, 0.9452, 0.9472, 0.9496 and 0.9522, respectively.

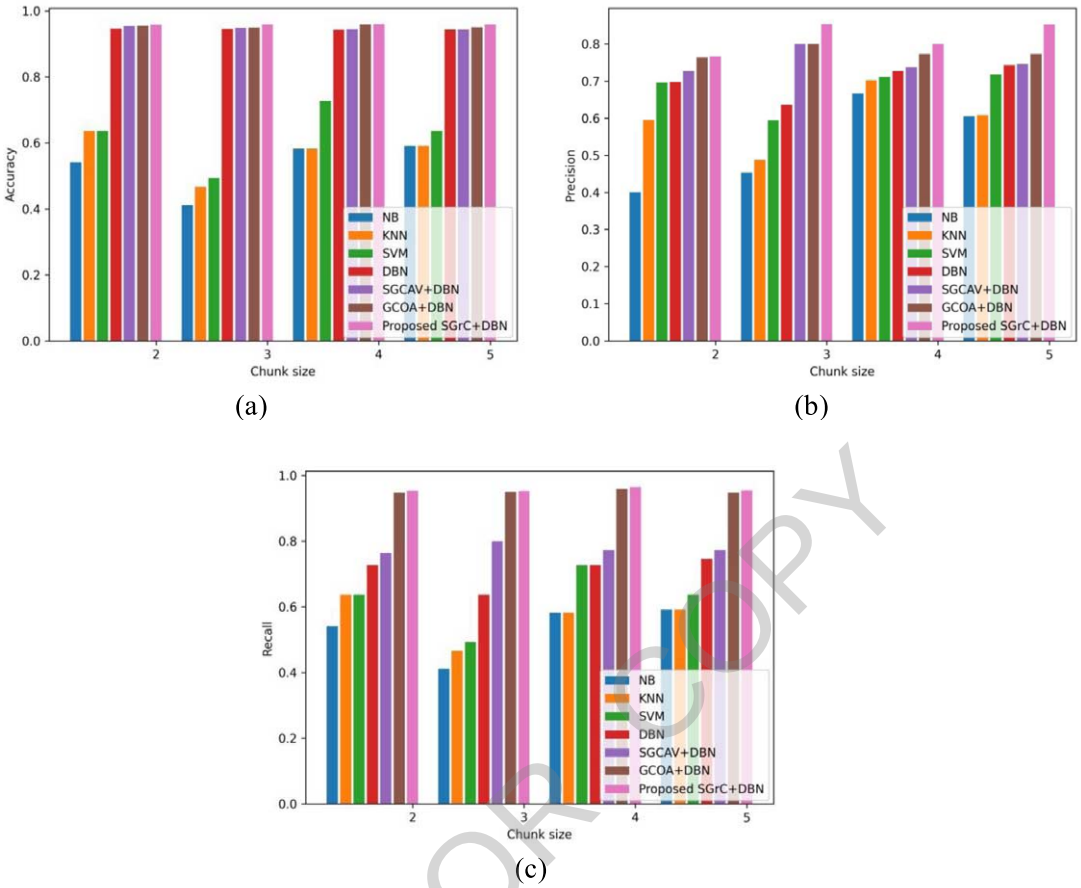


Fig. 5. Analysis of methods for feature size = 100 using (a) Accuracy (b) Precision (c) Recall.

Figure 6(b) depicts the analysis of the text classification methods based on precision. The precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 2 is 0.4641, 0.6132, 0.6383, 0.6818, 0.6909, 0.6964 and 0.7532, respectively. For chunk size = 5, the precision obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.6671, 0.6909, 0.7391, 0.7756, 0.8091, 0.8240 and 0.8571, respectively

Figure 6(c) depicts the analysis of the text classification methods based on recall. For chunk size = 2, the recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods are 0.5727, 0.6045, 0.6682, 0.6818, 0.6909, 0.9447 and 0.9491, respectively. The recall obtained by the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN and the proposed SGrC + DBN methods for chunk size = 5 is 0.6364, 0.6818, 0.6818, 0.6909, 0.8091, 0.9541 and 0.9581, respectively.

### *4.6. Comparative discussion*

Table 1 shows the comparative discussion of the text categorization method based on accuracy, precision and recall for the 20 newsgroup databases and Reuter databases. The maximum accuracy obtained by the proposed SGrC + DBN method is 0.9626, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the accuracy of 0.7618, 0.7798, 0.9375, 0.9416, 0.9538, and 0.9600 respectively for the 20 newsgroup databases. The proposed SGrC + DBN method obtained maximum precision of 0.9681, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the precision of 0.7500, 0.7900, 0.9048, 0.9427,

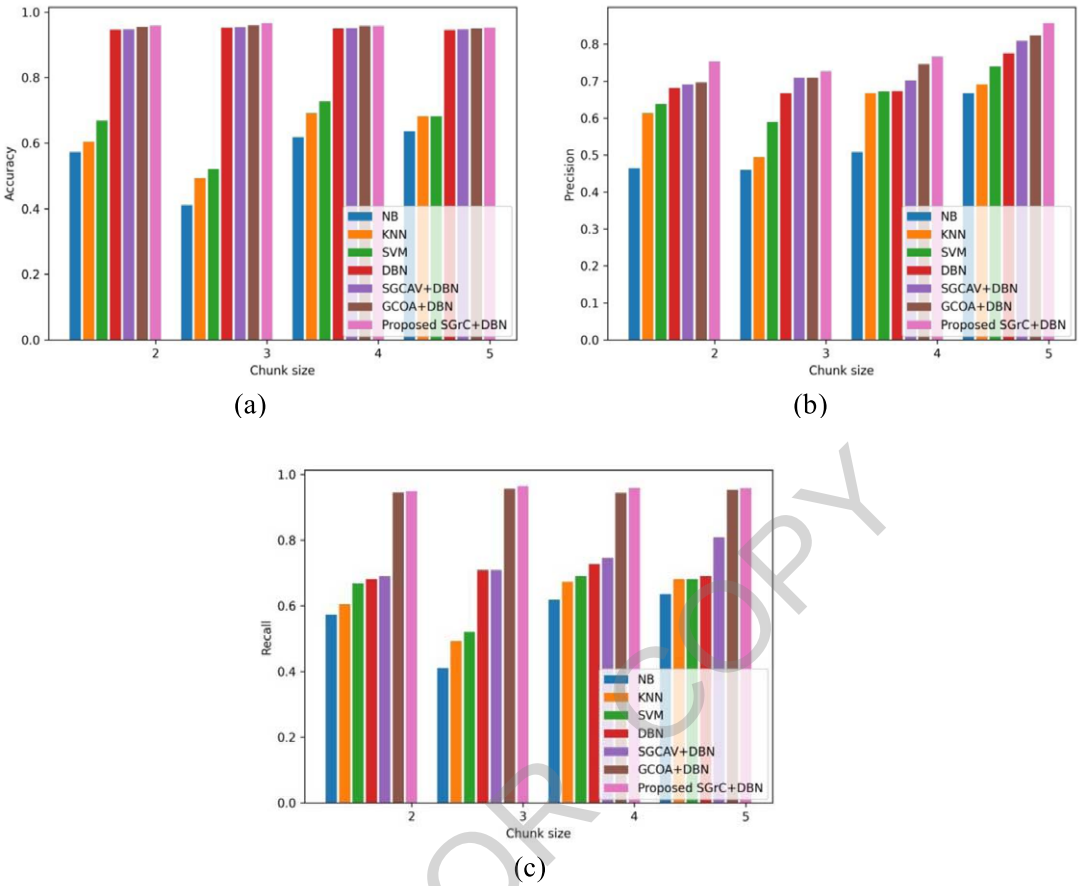


Fig. 6. Analysis of methods for feature size = 200 using (a) Accuracy (b) Precision (c) Recall.



0.9539, and 0.9596 respectively for the 20 newsgroup databases. The maximum recall obtained by the proposed SGrC + DBN method is 0.9600, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the precision of 0.7500, 0.7900, 0.9375, 0.9480, 0.9585, and 0.9590 respectively for the 20 newsgroup databases.

The maximum accuracy obtained by the proposed SGrC + DBN method is 0.9522, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the accuracy of 0.6364, 0.6818, 0.6818, 0.9452, 0.9472, and 0.9496 respectively for the Reuter databases. The proposed SGrC + DBN method obtained maximum precision of 0.8571, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the precision of 0.6671, 0.6909, 0.7391, 0.7756, 0.8091 and 0.8240 respectively for the Reuter databases. The maximum recall obtained by the proposed SGrC + DBN method is 0.9581, whereas the existing NB, KNN, SVM, DBN, SGCAV + DBN, GCOA + DBN obtained the precision of 0.6364, 0.6818, 0.6818, 0.6909, 0.8091, and 0.9541 respectively for the Reuter databases.

The reasons for the best performance of the proposed system are using stop word removal and stemming, VSM, information gain, and training the DBN using the proposed SGrC algorithm. Here, the redundant words in the text documents are removed by the stop word removal and stemming method during the pre-processing. Then, the TF-IDF features are extracted from the documents through the VSM for better categorization of the text. Also, the information gain is used for the selection of the relevant features for the categorization of text. Moreover, the proposed SGrC-based DBN offers accurate results for the text categorization. Thus, the proposed method attains

Table 1
Comparative discussion of the text categorization methods

| Database | Metric | NB | KNN | SVM | DBN | SGCAV + DBN | GCOA + DBN | Proposed SGrC + DBN |
|---|---|---|---|---|---|---|---|---|
| Using 20 newsgroup database | Accuracy | 0.7618 | 0.7798 | 0.9375 | 0.9416 | 0.9538 | 0.9600 | **0.9626** |
| | Precision | 0.7500 | 0.7900 | 0.9048 | 0.9427 | 0.9539 | 0.9596 | **0.9681** |
| | Recall | 0.7500 | 0.7900 | 0.9375 | 0.9480 | 0.9585 | 0.9590 | **0.9600** |
| Using Reuter database | Accuracy | 0.6364 | 0.6818 | 0.6818 | 0.9452 | 0.9472 | 0.9496 | **0.9522** |
| | Precision | 0.6671 | 0.6909 | 0.7391 | 0.7756 | 0.8091 | 0.8240 | **0.8571** |
| | Recall | 0.6364 | 0.6818 | 0.6818 | 0.6909 | 0.8091 | 0.9541 | **0.9581** |

better performance than the existing methods, such as NB, KNN, SVM, DBN, SGCAV + DBN, and GCOA + DBN, respectively.

## 5. Conclusion

In this research, an incremental text categorization method is developed using the proposed classifier. Initially, the redundant words from the document are removed through the pre-processing step. Then, the features are extracted from the pre-processed data using VSM. Once the features are extracted, the feature selection is done depending on the mutual information. These features altogether represent the feature vector and form the input to the text categorization model. The categorization of the text is done based on the proposed SGrC-based DBN. The proposed SGrC algorithm is the integration of the SMO algorithm and the GCOA algorithm, which trains the DBN classifier for the effective categorization of the text. Finally, the incremental text categorization is done using the hybrid weight bounding model that includes the Range degree and SGrC, and particularly, optimal weights for the Range degree model are selected using the SGrC algorithm. The experimental result of the proposed text categorization method is performed by considering the data from the Reuter database and 20 Newsgroups database. The comparative analysis of the text categorization methods is based on the performance metrics, such as precision, recall, and accuracy. When compared with the existing incremental text categorization methods, the proposed SGrC-based DBN obtained a maximum accuracy of 0.9626, maximum precision of 0.9681, and maximum recall of 0.9600, respectively.

In the future, the proposed system will be further extended for web page classification and email classification.



## Conflict of interest

None to report.

## References

[1] 20 Newsgroup database, http://qwone.com/~jason/20Newsgroups/, accessed on October 2018.
[2] M. Al-Diabat, Arabic text categorization using classification rule mining, *Applied Mathematical Sciences* **6**(81) (2012), 4033–4046.
[3] J. Chand Bansal, H. Sharma, S. Singh Jadon and M. Clerc, Spider Monkey Optimization algorithm for numerical optimization, *Memetic Computing* **6**(1) (2014), 31–47. doi:10.1007/s12293-013-0128-0.
[4] J. Chen, H. Huang, S. Tian and Y. Qu, Feature selection for text classification with naïve Bayes, *Expert Systems with Applications* **36**(3) (2009), 5432–5435. doi:10.1016/j.eswa.2008.06.054.
[5] N. Cheng, R. Chandramouli and K.P. Subbalakshmi, Author gender identification from text, *Digital Investigation* **8**(1) (2011), 78–88. doi:10.1016/j.diin.2011.04.002.
[6] G. D'Angelo and F. Palmieri, Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction, *Journal of Network and Computer Applications* **173** (2021).

[7] G. D'Angelo and S. Rampone, Diagnosis of aerospace structure defects by a HPC implemented soft computing algorithm, in: *Proceedings of the IEEE Conference on Metrology for Aerospace (MetroAeroSpace)*, 2014.
[8] S.J. Delany, M. Buckley and D. Greene, SMS spam filtering: Methods and data, *Expert Systems with Applications*, **39** (2012), 9899–9908.
[9] R. Elhassan and M. Ahmed, Arabic text classification review, *International Journal of Computer Science and Software Engineering (IJCSSE)* **4** (2015), 1.
[10] M. Ficco, F. Palmieri and A. Castiglione, Modeling security requirements for cloud-based system development, in: *Concurrency and Computation: Practice and Experience*, 2014.
[11] A.S. Ghareb, A.R. Hamdan and A.A. Bakar, Text associative classification approach for mining Arabic data set, in: *Data Mining and Optimization (DMO)*, 2012, pp. 114–120.
[12] M. Ghiassi, M. Olschimke, B. Moon and P. Arnaudo, Automated text classification using a dynamic artificial neural network model, *Expert Systems with Applications* **39**(12) (2012), 10967–10976. doi:10.1016/j.eswa.2012.03.027.
[13] G.E. Hinton, S. Osindero and Y. Teh, A fast learning algorithm for deep belief nets, *Neural Comput.* **18** (2006), 1527–1554. doi:10.1162/neco.2006.18.7.1527.
[14] M. Jiang, Y. Liang, X. Feng, X. Fan, Z. Pei, Y. Xue and R. Guan, Text classification based on deep belief network and softmax regression, *Neural Computing and Applications* **29**(1) (2016), 61–70. doi:10.1007/s00521-016-2401-x.
[15] T. Joachims, Text categorization with Support Vector Machines: Learning with many relevant features, in: *Machine Learning: ECML-98. ECML 1998*, C. Nédellec and C. Rouveirol, eds, Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), Vol. 1398, Springer, Berlin, Heidelberg, 1998.
[16] S.-B. Kim, K.-S. Han, H.-C. Rim and S. HyonMyaeng, Some effective techniques for navive Bayes text classification, *IEEE Transactions on Knowledge and Data Engineering* **18** (2006), 11.
[17] H. Li, B. Ma and C.H. Lee, A vector space modeling approach to spoken language identification, *IEEE Transactions on Audio, Speech, and Language Processing* **15**(1) (2007), 271–284. doi:10.1109/TASL.2006.876860.
[18] K. Luyckx and W. Daelemans, Personae: A corpus for author and personality prediction from text, in: *LREC*, 2008.
[19] T. Ma, G. Motta and K. Liu, Delivering real-time information services on public transit: A framework, *IEEE Transactions on Intelligent Transportation Systems* **18**(10) (2017), 2642–2656. doi:10.1109/TITS.2017.2656387.
[20] I. Maks and P. Vossen, A lexicon model for deep sentiment analysis and opinion mining applications, *Decision Support Systems* **53**(4) (2012), 680–688. doi:10.1016/j.dss.2012.05.025.
[21] C.D. Manning, C.D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
[22] A.H. Mohammad, T. Alwada'n and O. Al-Momani, Arabic text categorization using support vector machine, naïve Bayes and neural network, *GSTF Journal on Computing (JoC)* **5**(1) (2016), 108. doi:10.7603/s40601-016-0016-9.
[23] K. Motaz, Saad and W. Ashour, Arabic text classification using decision trees, in: *Proceedings of the 12th International Workshop on Computer Science and Information Technologies CSIT'2010*, Moscow–Saint-Petersburg, Russia, 2010.
[24] S.A. Özel, A web page classification system based on a genetic algorithm using tagged-terms as features, *Expert Systems with Applications* **38**(4) (2011), 3407–3415. doi:10.1016/j.eswa.2010.08.126.
[25] J.-Y. Park and J.-H. Kim, Incremental Class Learning for Hierarchical Classification, *IEEE Transactions On Cybernetics* (2018).
[26] N.M. Ranjan and R.S. Prasad, LFNN: Lion fuzzy neural network-based evolutionary model for text classification using context and sense based features, *Applied Soft Computing* (2018).
[27] Reuter database, https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection, accessed on October 2018.
[28] G. Sanghani and K. Kotecha, Incremental personalized email spam filter using novel TFDCR feature selection with dynamic feature update, *Expert Systems with Applications* **115** (2019), 287–299. doi:10.1016/j.eswa.2018.07.049.
[29] B. Scholkopf, K.K. Sung, C.J. Burges, F. Girosi, P. Niyogi, T. Poggio and V. Vapnik, Comparing support vector machines with Gaussian kernels to radial basis function classifiers, *IEEE transactions on Signal Processing* **45**(11) (1997), 2758–2765. doi:10.1109/78.650102.
[30] F. Sebastiani, Machine learning in automated text categorization' ACM publication, *ACM Computing Surveys* **34**(1) (2002), 1–47. doi:10.1145/505282.505283.
[31] G. Shan, S. Xu, L. Yang, S. Jia and Y. Xiang, Learn#: A Novel Incremental Learning Method for Text Classification, *Expert Systems with Applications* (2020), 113198. doi:10.1016/j.eswa.2020.113198.
[32] J.J. Sheu and K.T. Chu, An efficient spam filtering method by analyzing email's header session only, *International Journal of Innovative Computing, Information and Control* **5**(11) (2009), 3717–3731.
[33] S. Song, X. Qiao and P. Chen, Hierarchical text classification incremental learning, in: *Proceedings of International Conference on Neural Information Processing ICONIP, Neural Information Processing*, 2009, pp. 247–258.
[34] S.K. Srivastava, S.K. Singh and J.S. Suri, Effect of incremental feature enrichment on healthcare text classification system: A machine learning paradigm, *Computer methods and programs in biomedicine* **172** (2019), 35–51. doi:10.1016/j.cmpb.2019.01.011.
[35] J. Taeho, K nearest neighbor for text categorization using feature similarity, in: *ICAEIC-2019*, Vol. 2, 2019, p. 99.
[36] B. Tang, S. Kay and H. He, Toward optimal feature selection in naive Bayes for text categorization, *IEEE Transactions on Knowledge and Data Engineering* **28** (2016), 2508–2521. doi:10.1109/TKDE.2016.2563436.
[37] G. Toker and O. Kirmemis, Text categorization using k nearest neighbor classification, 2013.
[38] H. Uğuz, A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm, *Knowledge-Based Systems* **24**(7) (2011), 1024–1032. doi:10.1016/j.knosys.2011.04.014.

[39] J. Wohlwend, E.R. Elenberg, S. Altschul, S. Henry and T. Lei, Metric Learning for Dynamic Text Classification, 2019, arXiv preprint arXiv:1911.01026.
[40] J. Xu, C. Xu, B. Zou, Y.Y. Tang, J. Peng and X. You, New incremental learning algorithm with support vector machines, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **99** (2018), 1–12.
[41] Y. Yang and J.O. Pedersen, A comparative study on feature selection in text categorization, in: *Proceedings of International Conference on Machine Learning*, 1997, pp. 412–420.
[42] C. Yin and J. Xi, Maximum entropy model for mobile text classification in cloud computing using improved information gain algorithm, *Multimedia Tools and Applications* **76**(16) (2017), 16875–16891. doi:10.1007/s11042-016-3545-5.