Rider weed deep residual network-based incremental model for text classification using multidimensional features and MapReduce

Hemn Barzan Abdalla[1], Awder Ahmed[2], Subhi R. M. Zeebaree[3], Ahmed Alkhayyat[4], Baha Ihnaini[1]

[1] Department of Computer Science, Wenzhou-Kean University, Wenzhou, China
[2] Communication Engineering Department, Technical College of Engineering, Sulaimani Polytechnic University, Sulaimani, Iraq
[3] Culture Center, Duhok Polytechnic University, Duhok, Iraq
[4] College of Technical Engineering, the Islamic University, Najaf, Iraq
* Correspondence: habdalla@kean.edu

Abstract: Increasing demands for information and the rapid growth of big data have dramatically increased the amount of textual data. In order to obtain useful text information, the classification of texts is considered an imperative task. Accordingly, this article will describe the development of a hybrid optimization algorithm for classifying text. Here, pre-processing was done using the stemming process and stop word removal. Additionally, we performed the extraction of imperative features and the selection of optimal features using the Tanimoto similarity, which estimates the similarity between features and selects the relevant features with higher feature selection accuracy. Following that, a deep residual network trained by the Adam algorithm was utilized for dynamic text classification. Dynamic learning was performed using the proposed Rider invasive weed optimization (RIWO)-based deep residual network along with fuzzy theory. The proposed RIWO algorithm combines invasive weed optimization (IWO) and the Rider optimization algorithm (ROA). These processes are carried out under the MapReduce framework. Our analysis revealed that the proposed RIWO-based deep residual network outperformed other techniques with the highest true positive rate (TPR) of 85%, true negative rate (TNR) of 94%, and accuracy of 88.7%.

1. Introduction

The massive demand for big data has necessitated the evaluation of their sources and implications. The fundamental opinion of analysis relies on designing a novel framework for studying the data. The similarity measure is one of the mathematical models used for classifying and clustering data. Here, we provide the fundamental assessment of common similarity

Comment [MT1]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU2]: I agree with the change

measures such as Jaccard [10], cosine [12], Euclidean distance [9], and extended Jaccard [11], which are utilized for evaluating distance or angle across vectors. In this article, the similarity measures are divided into feature content or topological metrics.

In topology, the features are organized in a hierarchical model and the appropriate path length across the features must be evaluated. The features are measured based on evidence: features with elevated frequency have explicitly elevated information, while features with lower frequency are adapted with less information. Pair-wise and ITSim metrics fit into the class of feature content metrics. The information content measure provides elevated priority to the highest features with a small difference between the two data, leading to improved outcomes. The cosine and Euclidean belong to the class of topological metrics. It is susceptible to information loss as two similar datasets can be significantly offset by the existence of solitary features [4]. Methods such as clustering and classification that are utilized in text mining-based applications can also help transform massive data into small subsets to increase computational effectiveness [2].

Text data consist of noisy and irrelevant features that prevent learning techniques from improving their accuracy. To remove redundant data, various data mining methods have been adapted. Feature extraction and selection are two such methods used to classify data. The selection of components is utilized for eliminating extra text features for effective classification and clustering. The previous techniques transformed huge data into small data while taking classical distance measures into consideration. Reducing dimensionality minimizes evaluation time and maximizes the efficiency of classification. The recovery of data and text are utilized in detecting data synonyms and meaning. Several techniques have been devised for the classification and clustering processes. Clustering is carried out using unsupervised techniques with different class label data [2]. The goal of classifying text is to categorize data into different parts. In this study, the goal was to allocate pertinent labels based on content [3].

Categorizing texts is considered a crucial part of processing natural language. It is extensively employed in applications such as automatic medical text classification [35] and traffic monitoring [36]. Most news services require repeated arrangement of numerous articles in a single day [13]. Advanced email services offer the function of sorting junk mail and mail in an automated manner [14]. Other applications involve the analysis of sentiment [15], modeling topics [16], text clustering [32], translation of languages [17], and intent detection [18] [3]. Technology classification assists people, filters useful data, and poses more implications in real life. The design of text categorization and machine learning [38][37] has shifted from manual to machine [19] [20] [21] [22]. Several textualization classification techniques exist [23] with the goal of categorizing textual data. The categorization outcomes can fulfill an individual's requirements for classifying text and are suitable for rapidly attaining significant data. MapReduce is utilized for handling huge amounts of data using unstructured data [5]. Our aim is to devise an optimization-driven deep learning technique for classifying texts using the MapReduce framework. First, the text data underwent pre-processing to remove unnecessary words. Pre-processing was performed using stop word removal and the stemming process. After that, we

Comment [MT3]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU4]: I agree with the change

Comment [MT5]: Do you mean "two datasets"?

Comment [WU6]: Between the data

Comment [MT7]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU8]: I agree with the change

Comment [MT9]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU10]: I agree with the change

extracted the features, such as SentiWordNet, thematic, and contextual features. These features were employed in a deep residual network for classifying the texts and the deep residual network training was performed using the Adam algorithm. Finally, dynamic learning was carried out wherein the proposed Rider invasive weed optimization (RIWO)-based deep residual network was employed for incremental text classification. The fuzzy theory was employed for weight bounding to deal with the incremental data. In this process, the deep residual network training was performed using the proposed RIWO, which was devised by combining the Rider optimization algorithm (ROA) and invasive weed optimization (IWO) algorithm.

The key contributions of this paper are:

- Proposed RIWO-based deep residual network for text classification: A new method developed using multidimensional features and MapReduce. Dynamic learning uses the proposed RIWO-based deep residual network for classifying texts. Here, the developed RIWO was adapted for deep residual network training.

- RIWO: Devised by combining ROA and IWO algorithms.

- The fuzzy theory: Employed to handle dynamic data by performing weight bounding.

The rest of the sections are given as follows: Section 2 presents the classical text classification techniques survey. Section 3 describes the developed text classification model. Section 4 discusses the results of the developed model for classical techniques, and section 5 presents the conclusion.

2. Literature review

The eight classical techniques based on text classification using big data and their issues are described below. Ranjan & Prasad [1] developed an LFNN-based incremental learning technique for classifying text data based on context-semantic features. The methods employed a dynamic dataset for classification to dynamically learn the model. Here, we employed the incremental learning procedure Back Propagation Lion (BPLion) Neural Network, and fuzzy bounding and Lion Algorithm (LA) were used to select the weights. However, the technique failed to precisely classify the sentence. Kotte et al. [2] devised a similarity function for clustering the feature pattern. The technique attained dimensionality reduction with improved accuracy. However, the technique failed to utilize membership functions for obtaining clusters. Wang et al. [3] devised a deep learning technique for classifying text documents. Additionally, a large-scale scope-based convolutional neural network (LSS-CNN) was utilized for categorizing the text. The method effectively computed scope-based data and parallel training for massive datasets. The technique attained high   scalability on big data but failed to attain the utmost accuracy. Kuppili et al. [4] developed the Maxwell–Boltzmann Similarity Measure (MBSM) for classifying text. The MBSM was derived with feature values from the documents. The MBSM was devised by combining single label K-nearest neighbor's classification (SLKNN), multi-label KNN (MLKNN), and K-means clustering. However, the technique failed to include clustering techniques and query mining. Liu & Wang [5] devised a technique for classifying text using English quality-related text data. Here, the goal was to extract, classify, and examine the data from

English texts while considering cyclic neural networks. Ultimately, the features with sophisticated English texts were generated. This technique also combined attention to improve label disorder and make the structure more reliable. However, the computation cost tended to be very high. Li et al. [6] designed a method for classifying text and solving the misfitting issue by performing angle-pruning tasks from a database. The technique computed the efficiency of each convolutional filter using discriminative power produced at the pooling layer and shortened words obtained from the filter. However, the technique produced high computational complexity.

Bensaid & Alimi [7] devised the Multi-Objective Automated Negotiation-based Online Feature Selection (MOANOFS) for classifying texts. The MOANOFS utilized automated negotiation and machine learning techniques to improve classification performance using ultra-high dimensional datasets. This helped the method to decide which features were the most pertinent. However, the method failed to select features from multi-classification domains. Jiang et al. [8] devised a hybrid text classification model based on softmax regression for classifying text. The deep belief network was utilized to classify text using learned feature space. However, the technique failed to filter extraneous characters for enhancing system performance.

Akhter et al. [39] developed a large multi-format and multi-purpose dataset with more than ten thousand documents organized into six classes. For text classification, they utilized a Single-layer Multisize Filters Convolutional Neural Network (SMFCNN). The SMFCNN obtained high accuracy, demonstrating its capability to classify long text documents in Urdu. Flores et al. [40] developed a query strategy and stopping criterion that transformed Classifier Regular Expression (CREGEX) in an active learning (AL) biomedical text classifier. As a result, the AL was permitted to decrease the number of training examples required for a similar performance in every dataset compared to passive learning (PL).

Huan et al. [41] introduced a method for Chinese text classification that depended on a feature-enhanced nonequilibrium bidirectional long short-term memory (Bi-LSTM) network. This method enhanced the precision of Chinese text classification and had a reliable capability to recognize Chinese text features. However, the accuracy of Chinese text recognition needs improvement and the training processing time should be reduced. Dong et al. [42] introduced a text classification approach using a self-interaction attention mechanism and label embedding. This method showed high classification accuracy, but for practical application, more work should be done.

3. Proposed RIWO-based deep residual network for text classification in big data

The objective of text classification is to categorize text data into different classes based on certain content. Text classification is considered an imperative part of processing natural language. However, it is considered a challenging and complex process due to high dimensional and noisy texts, and the need to devise an improved classifier for huge textual data. This study devised a novel hybrid optimization-driven deep learning technique

188   for text classification using big data. Here, the goal was to devise a classifier
189   that employs text data as input and allocates pertinent labels based on
190   content. At first, the input text data underwent pre-processing to eliminate
191   noise and artifacts. Pre-processing was performed with stop word removal
192   and stemming. Once the pre-processed data were obtained, the contextual,
193   thematic, and SentiWordNet features were extracted. Once the features were
194   extracted, the imperative features were chosen using the Tanimoto similarity.
195   The Tanimoto similarity method evaluates similarity across features and
196   chooses the relevant features with high feature selection accuracy. Once the
197   features were selected, a deep residual network [26] was used for dynamic
198   text classification. The deep residual network was trained using the Adam
199   algorithm [11] [33] [34]. Additionally, dynamic learning was performed using
200   the proposed RIWO algorithm along with the fuzzy theory. The proposed
201   RIWO algorithm integrates IWO [27] and ROA [28]. Figure 1 shows the
202   schematic view of text classification from the input text data in big data using
203   the proposed RIWO method, considering the MapReduce phase.

204
205

206   **Figure 1.** Schematic view of text classification from the input big data using the
207   proposed RIWO-based deep residual network.

208   Assume input text data with various attributes is expressed as:

$$B = \{B_{d,e}\}; (1 \le d \le D)(1 \le e \le E) \tag{1}$$

209   Where, $B_{d,e}$ refers to text data contained in the database with an attribute
210   in data. Data points are employed using attributes for each data point. The
211   other step was to eliminate artifacts and noise present in the data.
212   The data in a database are split into a specific number that is equivalent
213   to mappers present in the MapReduce model. The partitioned data is given
214   by:

$$B_{d,e} = \{D_q\}; 1 \le q \le N \tag{2}$$

215   Where, N symbolizes total mappers. Assume mappers in MapReduce
216   are expressed as:

$$M = \{M_1, M_2, \square\ , M_q, \square\ , M_N\}; 1 \le q \le N \tag{3}$$

217   Thus, input to mapper is formulated as:

$$D_q = \{d_{r,l}\}; 1 \le r \le m_q ; 1 \le l \le n \tag{4}$$

218   Where $d_{r,l}$ symbolizes split data given to mapper, and D_q indicates data
219   in mapper.

220   *3.1. Pre-processing*

221   The partitioned data from the text dataset was pre-processed by
222   removing stop words and using stemming. Pre-processing is an important
223   process used to smoothly arrange various data and offer effective outcomes
224   by improving representation. The dataset contains unnecessary phrases and

words that influence the process. Therefore, pre-processing is important for removing inconsistent words from the dataset. Initially, the text data are accumulated in the relational dataset and all reviews are divided into sentences and bags of the sentence. The elimination of stop words is carried out to maximize the performance of the text classification model. Here, the stemming and stop word removal refined the data.

3.1.1. Stop word removal

This is a process that removes words with less representative value for the data. Some of the non-representative words include pronouns and articles. When evaluating data, some words are not valuable to text content, and removing such redundant words is imperative. This procedure is termed stop word removal [29]. Certain words such as articles, conjunctions, and prepositions, continuously appear and are called stop words. The removal of the stop word, the most imperative technique, is utilized to remove redundant words using vocabulary because the vector space size does not offer any meaning. The stop words indicate the word, which does not hold any data. It is a process used to eliminate stop words from a large set of reviews. The elimination of the stop word is used to save space and accelerate and improve processing.

3.1.2. Stemming

The stemming procedure is utilized to convert words to stem form. In massive amounts of data, several words are utilized that convey a similar meaning. Therefore, the critical method used to minimize words to root is stemming. Stemming is a method of linguistic normalization wherein little words are reduced. Moreover, it is the procedure used to retrieve information for describing the mechanism of reducing redundant words to their root form and word stem. For instance, the words connections, connection, connecting, and connected are all reduced to connect [29].

$$M_l = \left\{ Q_i, \ 1 \le \eta \le P_k \right\}$$

Where Q_i symbolizes total words present in text data from the database.

The pre-processed outcome generated from pre-processing is expressed as M_l, which is subjected as an input to feature extraction phase.

3.2 Acquisition of features for producing highly pertinent features

This describes an imperative feature produced with input review, and the implication of feature extraction is used to produce pertinent features that facilitate improved text classification. Moreover, data obstruction is reduced because text data is expressed as a minimized feature set. Therefore, the pre-processed partitioned data is fed to feature extraction, wherein SentiWordNet, contextual, and thematic features are extracted.

3.2.1. Extraction of SentiWordNet features

The SentiWordNet features are utilized from pre-processed partitioned data by removing keywords from reviews. Here, the SentiWordNet [31] is employed as a lexical resource to extract the SentiWordNet features. The SentiWordNet assigns each WordNet text one of three numerical sentiment scores: positive, negative, or neutral. Here, different words indicated

Comment [MT11]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU12]: I agree with the change

Comment [MT13]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU14]: I agree with the change

different polarities that indicated various word senses. The SentiWordNet consisted of different linguistic features: verbs, adverbs, adjectives, and n-gram features. SentiWordNet is a process used to evaluate the score of a specific word using text data. Here, the SentiWordNet was employed to determine the polarity of the offered review and for discovering positivity and negativity. Hence, SentiWordNet is modeled as F_1.

3.2.2. Extraction of contextual features

The context-based features [1] were generated from pre-processed partitioned data that described relevant words by dividing them using non-relevant reviews for effective classification. This requires finding key terms that have context and semantic meaning in order to establish a proper context. The key term is considered a preliminary indicator for relevant review while context terms act as a validator that can be used to evaluate if the key term is an indicator. Here, the training dataset contained keywords with pertinent words. The context-based features assisted in selecting the relevant and non-relevant reviews.

We considered representing a training dataset that had relevant and non-relevant reviews. Using this method, assume x_s represents the key term and x_c indicates the context term.

- Detection of key terms:

The language model that employs each term and the metric are expressed as:

$$C = \frac{L_{rel}}{L_{on\_rel}}$$

Where, L_{rel} symbolizes the language model for N_{rel} and $L_{non-rel}$ signifies the language model for N_{non_rel}.

- Discovery of the context term:

After discovering key terms, the process of context term discovery, which is similar to separately detecting each term, begins. The steps employed in determining the context term are given as:

i) Computing all instances of the key term employed among relevant and non-relevant reviews.

ii) By employing sliding window size, the conditions are mined as context terms. Hence, the size of the window is employed as a context span.

iii) The pertinent terms generated are employed as a text, modeled as d_r, and non-relevant terms are denoted as d_{nr}. The set of pertinent text is modeled as R_{d_r}, and the non-relevant set is referred to as R_{d_nr}.

iv) After that, the score evaluated for each distinctive term is expressed as:

$$C(x_C) = | L_{R_{d\_r}}(x_C) - L_{R_{d\_nr}}(x_C) |_S \qquad (7)$$

Where $L_{R_{d\_r}}(x_C)$ symbolizes the language model for an excerpt with relevant review set. The term $L_{R_{d\_nr}}(x_C)$ indicates a language model for an

Comment [MT15]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU16]: I agree with the change

Comment [MT17]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU18]: I agree with the change

excerpt with a non-relevant review set and S represents the size of the window. If the measure is a definite threshold then that score is adapted as a context term x_S. Generated context-based features are modeled as F_2.

3.2.3. Extraction of thematic features

The pre-processed partitioned data $d_{r,l}$ is used to find thematic features. Here, the count of the thematic word [30] in a sentence is imperative as the frequently occurring words are most likely connected to the topic in the data. Thematic words are words that grab key topics defined in a provided document. In thematic features, the top 10 most frequent words are employed as thematic. Thus, the thematic feature F_3 is modeled as:

$$F_1 = \frac{T}{\sum_{t=0}^{\square} T_t}$$

(8)

Where T expresses the count of thematic words in a sentence, and it is expressed as $\{T_1, T_2, \square, T_\square\}$.

The feature vectors considering the contextual, thematic, and SentiWordNet features are expressed as:

$$F = \{F_1, F_2, F_3\}$$

(9)

Where F_1 symbolizes SentiWordNet features, F_2 signifies contextual features, and F_3 refers to thematic features.

3.3. Feature selection using the Tanimoto similarity

The selection of imperative features from the extracted features F is made using the Tanimoto similarity. The Tanimoto similarity computes similarity across features and selects features with high feature selection accuracy. Here, the Tanimoto similarity is expressed as:

$$S = \frac{\sum_{w=1}^{m} y_w z_w}{\sum_{w=1}^{m} y_w^2 + \sum_{w=1}^{m} z_w^2 - \sum_{w=1}^{m} y_w z_w}$$

(10)

Where S indicates the Tanimoto measure y_w and z_w represents features. The selected features are expressed as R.

The produced feature selection output obtained from the mapper is input to the reducer U. Then, the text classification is performed on the reducer using the selected features, which is briefly illustrated below.

3.4. Classification of texts with Adam-based deep residual network

Text classification is performed using an Adam-based deep residual network and selected features R. The classification of text data assists in standardizing the infrastructure and makes the search simpler and more pertinent. Additionally, classification enhances the user's experience, simplifies navigation, and helps solve business issues (such as social media and e-mails) in real-time. The deep residual network is more effective at counting attributes and computation. This network is capable of building deep representations at each layer and can manage advanced deep learning tasks. The architecture of the deep residual network and training with the Adam algorithm is described below.

3.4.1. Architecture of the deep residual network

We employed a deep residual network [26] in order to make a productive decision regarding which text classification to perform. The DRN is comprised of different layers: residual blocks, convolutional (Conv) layers, linear classifier, and average pooling layers. Figure 2 shows the structural design of a deep residual network with residual blocks, Conv layers, linear classifier, and average pooling layers for text classification.

-Conv layer:

The two-dimensional Conv layer reduces free attributes in training and offers reimbursement for allocating weight. The cover layer processes the input image with the filter sequence known as the kernel using a local connection. The cover layer utilizes a mathematical process for sliding the filter with the input matrix and computes the dot product of the kernel. The evaluation process of the Conv layer is represented as:

$$B2d(O) = \sum_{a=0}^{E-1}\sum_{s=0}^{E-1} X_{a,s} \bullet O_{(u+a),(v+s)} \tag{11}$$

$$B1d(O) = \sum_{Z=0}^{C_{in}-1} G_Z * O \tag{12}$$

Where O expresses the CNN feature of the input image, u and v refer to the recording coordinates, G signifies the $E \times E$ kernel matrix termed as a learnable parameter, and a and s are the position indices of the kernel matrix. Hence, G_z expresses the size of the kernel for the Zth input neuron and $*$ expresses the cross-correlation operator.

Pooling layer: This layer is associated with the Conv layer and is especially utilized to reduce the feature map's spatial size. The average pooling is selected as a function of each slice and the depth of the feature map.

$$a_{out} = \frac{a_{in} - Z_a}{\lambda} + 1$$

$$s_{out} = \frac{s_{in} - Z_s}{\lambda} + 1$$

Where a_{in} symbolizes the input matrix width, s_{in} signifies the height of the input matrix, a_{out} and s_{out} represent the respective value of output, and Z_a and Z_s symbolize the width and height of the kernel size.

Comment [MT19]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU20]: I agree with the change

375 -Activation function: The nonlinear activation function is adapted for
376 learning nonlinear and complicated features so it is utilized to improve the
377 non-linearity of extracted features. Rectified linear unit (ReLU) is utilized for
378 processing data. The ReLU function is formulated as:

$$\mathrm{Re}\,LU(O)=\begin{cases}0 \; ; \; \mathrm{K} < 0 \\ \mathrm{K} \; ; \; \mathrm{K} \geq 0\end{cases} \tag{15}$$

379 Where K symbolizes a feature.
380 -Batch normalization: Here, the training set was divided into various
381 small sets known as mini-batches to train the model. It attains a balance
382 between evaluation and convergence complexity. The input layers are
383 normalized by scaling activations to maximize reliability and training speed.
384 -Residual blocks: This indicates the shortcut connection amongst the
385 Conv layers. The input is unswervingly allocated to output only if input and
386 output are of equal size.

$$\kappa = \Re(O) + O \tag{16}$$

$$\kappa = \Re(O) + \mathrm{D}_M\,O \tag{17}$$

387 Where O and κ signify input and output residual blocks, κ
388 symbolizes mapping relation, D_M expresses dimension matching factor, and
389 $\Re(.)$ signifies activation function.
390 -Linear classifier: After completion of the Conv layer, linear classifier
391 performs a procedure to discover noisy pixels using input features. It is a
392 combination of the softmax function and a fully connected layer.

$$\kappa = \mathrm{D}\kappa + \upsilon \tag{18}$$

393 Where D expresses weight matrix and υ represents bias. Figure 2 shows
394 the structural design of the deep residual network. Here, the output,
395 represented as κ, assists in classifying the texts.
396
397

398 **Figure 2.** Structural design of the deep residual network with residual blocks,
399 convolutional (Conv) layers, linear classifier, and average pooling layers for text
400 classification.

401 3.4.2. Training of the deep residual network with the Adam algorithm

402 The deep residual network training is performed using the Adam
403 technique which assists in discovering the best weights for tuning the deep
404 residual network for classifying text. Adam [11] represents a first-order
405 stochastic gradient-based optimization extensively adapted to a fitness
406 function that changes for attributes. The major implication of the method is
407 computational efficiency and fewer memory needs. Moreover, the problems
408 associated with the non-stationary objectives and the subsistence of noisy
409 gradients are handled effectively. In this study, the magnitudes of the
410 updated parameters were invariant in contrast to the rescaling of gradient,
411 and step size was handled with a hyperparameter that worked with sparse
412 gradients. In addition, Adam is effective in performing step size annealing.
413 The classification of text employs a deep residual network for texts. The steps
414 of Adam are given as:

415   Step 1: Initialization

416   The first step represents bias correction initialization wherein $\hat{q}_l$ signifies

417   the corrected bias of the first moment estimate and $\hat{m}_l$ represents the
418   corrected bias of the second moment estimate.
419   Step 2: Discovery of error
420   The bias error is computed to choose the optimum weight for training
421   the deep residual network. Here, the error was termed as an error function
422   that led to an optimal global solution. The function is termed as a
423   minimization function and is expressed as:

$$Err = \frac{1}{f}\sum_{l=1}^{f}\left(O_l - \kappa\right)^2 \tag{19}$$

424   Where f signifies total data, κ symbolizes output generated with the

425   deep residual network classifier, and O_l indicates the expected value.
426   Step 3: Discovery of updated bias
427   Adam is used to improving convergence behavior and optimization.
428   This technique generates smooth variation with effectual computational
429   efficiency and lower memory requirements. As per Adam [11], the bias is
430   expressed as:

$$\theta_l = \theta_{l-1} - \frac{\alpha\hat{q}_l}{\sqrt{\hat{m}_l + \varepsilon}} \tag{20}$$

431   Where α refers to step size, $\hat{q}_l$ expresses corrected bias, $\hat{m}_l$ indicates bias-

432   corrected second-moment estimate, ε represents the constant, and θ_{l-1}
433   signifies the parameter at a prior time instant $(l-1)$. The corrected bias of the
434   first-order moment is expressed as:

$$\hat{q}_l = \frac{q_l}{(1-\eta_1^l)} \tag{21}$$

$$\hat{q}_l = \eta_1 q_{l-1} + (1-\eta_1)G_l^1 \tag{22}$$

435   The corrected bias of the second-order moment is represented as:

$$\hat{m}_l = \frac{m_l}{(1-\eta_2^l)} \tag{23}$$

$$\hat{m}_l = \eta_2 m_{l-1} + (1-\eta_2)H_l^2 \tag{24}$$

$$H_l = \nabla_\theta loss(\theta_{l-1}) \tag{25}$$

436   Step 4: Determination of the best solution: The best solution is
437   determined with error, and a better solution is employed for classifying text.
438   Step 5: Termination: The optimum weights are produced repeatedly
439   until the utmost iterations are attained. Table 1 describes the pseudocode of
440   the Adam technique.

441   **Table 1.** Pseudocode of the Adam algorithm.

442

443   *3.5. Dynamic learning with the proposed RIWO-based deep residual network*

For incremental data B, dynamic learning is done using the proposed RIWO-based deep residual network. Here, the assessment of incremental learning with the developed RIWO-based deep residual network was done to achieve effective text classification with the dynamic data. The deep residual network was trained with developed RIWO for generating optimum weights. The developed RIWO was generated by integrating ROA and IWO for acquiring effective dynamic text classification.

3.5.1. Architecture of deep residual network

The model of the deep residual network is described in section 3.4.

3.5.2. Training of deep residual network with proposed RIWO

The training of the deep residual networks was performed with the developed RIWO, which was devised by integrating IWO and ROA. Here, the ROA [28] was motivated by the behavior of the rider groups, which travel and compete to attain a common target position. In this model, the riders were chosen from the total number of riders for each group. We concluded that this method produces enhanced classification accuracy. Furthermore, the ROA is effective and follows the steps of fictional computing for addressing optimization problems but with less convergence. IWO [27] is motivated by colonizing characteristics of weed plants. The technique showed a fast convergence rate and elevated the accuracy. Hence, we integrated IWO and ROA to enhance complete algorithmic performance. The steps in the method are expressed as:

Step 1) Initialization of population

The preliminary step is algorithm initialization, which is performed using four-rider groups provided by A and represented as:

$$A = \{A_1, A_2, \square \; A_\mu, \square \; , A_\vartheta\} \tag{26}$$

Where A_μ signifies μ^{th} rider and ϑ is the total riders.

Step 2) Determination of error:

The computation of errors is already described in equation (19).

Step 3) Update the riders' position:

The rider position in each set is updated for determining the leader. Thus, the updated rider position using a feature of each rider is defined below. The updated position of each rider is expressed as:

As per ROA [28], the updated overtaker position is used to increase the rate of success by determining the position of the overtaker and is represented as:

$$A_{n+1}^o(g,h) = A_n^o(g,h) + \left[\partial_n^*(g) * A^L(L,h)\right] \tag{27}$$

Where $\partial_n^*(g)$ signifies the direction indicator.

The attacker has a propensity to grab the position of the leaders and is given by:

$$A_{n+1}^a(g,u) = A^L(L,u) + \left[Cos \, K_{g,u}^n * A^L(L,u)\right] + r_g^n \tag{28}$$

The bypass riders contain a familiar path, and its update is expressed as:

$$A_{n+1}^b(g,u) = \lambda\left[A_n(\chi,u) * \delta(u) + A_n(\xi,u) * \left[1 - \delta(u)\right]\right] \tag{29}$$

Comment [MT21]: Please confirm if the changes made conform to the intended meaning of the sentence.

Comment [WU22]: I agree with the change

483   Where λ symbolizes the random number, χ signifies the arbitrary

484   number between 1 and P, ξ denotes an arbitrary number between 1 and P,

485   and δ expresses an arbitrary number between 0 and 1.

486   The follower poses a propensity to update the position using the leading

487   rider position to attain the target and is given by:

$$A_{n+1}^{F}(g,h) = A^{L}(L,h) + \left[Cos(K_{g,h}^{n} * A^{L}(L,h) * r_{g}^{n})\right] \tag{30}$$

488   Where h is the coordinate selector, A^{L} indicates the leading rider

489   position, L represents the leading rider index, $K_{g,h}^{n}$ represents the steering

490   angle of gth rider in h^{th} coordinate, and r_{g}^{n} is the distance.

$$A_{n+1}^{F}(g,h) = A^{L}(L,h)\left[1 + Cos(K_{g,h}^{n} * r_{g}^{n})\right] \tag{31}$$

491   The IWO assists in generating the best solutions. Per IWO [27], the

492   equation is represented as:

$$A_{n+1}^{F}(g,h) = \sigma(n)A_{n}^{F} + A_{best} - A_{n}^{F} \tag{32}$$

493   Where A_{n+1}^{F} symbolizes the new weed position in iteration $n+1$, A_{n}^{F}

494   signifies the current weed position, A_{best} refers to the best weed found in the

495   whole population, and $\sigma(n)$ represents the current standard deviation.

$$A_{best} = A_{n+1}^{F}(g,h) - \sigma(n)A_{n}^{F} + A_{n}^{F} \tag{33}$$

496   Substitute equation (33) in equation (31),

$$A_{n+1}^{F}(g,h) = \left(A_{n+1}^{F}(g,h) - \sigma(n)A_{n}^{F} + A_{n}^{F}\right)\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] \tag{34}$$

$$A_{n+1}^{F}(g,h) = A_{n+1}^{F}(g,h)\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] - \left(\sigma(n)A_{n}^{F} + A_{n}^{F}\right)\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] \tag{35}$$

$$A_{n+1}^{F}(g,h) - A_{n+1}^{F}(g,h)\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] = \left[A_{n}^{F} - \sigma(n)A_{n}^{F}\right]\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] \tag{36}$$

$$A_{n+1}^{F}(g,h)\left[1 - 1 - Cos(K_{g,h}^{n}) * r_{g}^{n}\right] = \left[A_{n}^{F} - \sigma(n)A_{n}^{F}\right]\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] \tag{37}$$

$$A_{n+1}^{F}(g,h)\left[- Cos(K_{g,h}^{n}) * r_{g}^{n}\right] = \left[A_{n}^{F} - \sigma(n)A_{n}^{F}\right]\left[1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right] \tag{38}$$

497   The final updated equation of the proposed RIWO is expressed as:

$$A_{n+1}^{F}(g,h) = -\frac{(A_{n}^{F} - A_{n}^{F}\sigma(n)\left(1 + Cos(K_{g,h}^{n}) * r_{g}^{n}\right)}{Cos(K_{g,h}^{n}) * r_{g}^{n}} \tag{39}$$

498   Step 4) Re-evaluation of the error:

499   After completing the update process, the error of each rider is computed.

500   The position of the rider in the leading position is replaced using the position

501   of the new generated rider so that the error of the new rider is smaller.

502   Step 5) Update of the rider parameter:

503   The rider attribute update is imperative to determine an effectual

504   optimal solution using the error.

505   Step 6) Riding off time:

506   The steps were iterated repeatedly until we attained off time N_{OFF}, in

507   which the leader was determined. The pseudocode of the developed RIWO

508   is shown in Table 2.

509   **Table 2.** Pseudocode of the developed RIWO.

The output produced from the developed RIWO-based deep residual network is κ, which helps classify the text data since dynamic learning helps classify the dynamic data. Here, we employed fuzzy bounding to remodel the classifier if there was a high chance of a previous data error.

3.5.3. Fuzzy theory

An error is evaluated whenever incremental data is added to the model and weights are updated without using the previous weights. If the error evaluated by the present instance is less than the error of the previous instance then the weights are updated based on the proposed RIWO algorithm. Otherwise, the classifiers are remodeled by setting a boundary for weight using fuzzy theory [1] and optimal weight is chosen using the proposed RIWO algorithm. On arrival of data d_{i+1}, the error e_{i+1} will be computed and compared with that of the previous data d_i. If $e_i < e_{i+1}$, then prediction with training based on RIWO is made. Otherwise, the fuzzy bounding-based learning will be done by bounding the weights, which is given as:

$$\omega^B = \omega^t \pm F^s \tag{40}$$

Where ω^t is weight at the current iteration and F^s signifies a fuzzy score. For the dynamic data, the features $\{F\}$ were extracted. Here, the membership degree is given as:

$$Membership\ Degree = \| \omega^{t-2} - \omega^{t-1} \| \tag{41}$$

Where ω^{t-2} represents weights at iteration $t-2$ and ω^{t-1} signifies weights at iteration $t-1$. When the highest iteration is attained, the process is stopped.

4. Results and Discussion

The competence of the technique is evaluated by analyzing the techniques using various measures like true positive rate (TPR), true negative rate (TNR), and accuracy. The assessment is done by considering mappers=3, mappers=4, and by varying the chunk size.

4.1 Experimental setup

The execution of the developed model was performed in PYTHON with Windows 10 OS, an Intel processor, and 4GB RAM. Here, the analysis was performed by considering the NSL-KDD dataset.

4.2. Dataset description

The dataset adapted for text classification involved the Reuters and 20 Newsgroups databases and is explained below.

4.2.1. 20 Newsgroups database

The 20 Newsgroups dataset [24] was curated by Ken Lang for newsreaders to extract the Netnews. The dataset was established by collecting 20,000 newsgroup data points split across 20 different newsgroups.

Comment [MT23]: Please confim if the changes made conform to the intended meaning of the sentence.

Comment [WU24]: I agree with the change

549   The database is popular for analyzing text applications used to handle
550   machine-learning methods such as clustering and text classification. The
551   dataset is organized into 20 different newsgroups covering different topics.

552   4.2.2. Reuters database

553   The Reuters-21578 Text Categorization Collection Dataset was curated
554   by David D. Lewis [25]. The dataset is comprised of documents collected from
555   Reuters newswires starting in 1987. The documents are arranged and indexed
556   based on categories. There were 21,578 instances in the dataset with five
557   attributes. The number of websites attained by the dataset was 163,417.

> **Comment [MT25]:** Please confirm if the changes made conform
>
> **Comment [WU26]:** I agree with the change

558   *4.3 Evaluation metrics*

559   The efficiency of the developed model was examined by adopting
560   measures such as accuracy, TPR, and TNR.

561   4.3.1. Accuracy

562   Accuracy is described as the measure of data that is precisely preserved
563   and is expressed as:

$$Acc = \frac{P+Q}{P+Q+H+F} \tag{42}$$

564   Where P signifies true positive, Q symbolizes true negative, H denotes
565   true false positive, and F is a false negative.

566   4.3.2. TPR

567   The TPR refers to the ratio of the count of true positives with respect to
568   the total number of positives.

$$TPR = \frac{P}{P+H} \tag{43}$$

569   Where P refers to true positives and H is the false negatives.

570   4.3.3. TNR

571   The TNR refers to the ratio of negatives that are correctly detected.

$$TNR = \frac{Q}{Q+F} \tag{44}$$

572   Where Q is true negative and F signifies false positive.

573   *4.4. Comparative methods*

574   We evaluated the proposed RIWO-based deep residual network by
575   comparing it with other classical techniques such as LSS-CNN [3], RNN [5],
576   SLKNN+MLKNN [4], BPLion+LFNN [1], SVM, NN, and LSTM.

> **Comment [MT27]:** Please confirm if the changes made conform to the intended meaning of the sentence.
>
> **Comment [WU28]:** I agree with the change

577   *4.5. Comparative analysis*

578   The proposed technique was assessed using certain measures such as
579   accuracy, TPR, and TNR. Here, the analysis was performed by considering
580   the Reuters and 20 Newsgroups datasets, as well as the mapper size=3 and 4.

581   4.5.1. Analysis with the Reuters dataset

582   The assessment of techniques with the Reuters dataset considering TPR,
583   TNR, and accuracy parameters is described. The assessment is done with
584   mapper=3 and mapper=4 and varying the chunk size.

585
586   a) Assessment with mapper=3
587

588   **Figure 3.** Assessment of different techniques, comparing the proposed method with
589   the Reuters dataset with mapper=3 **a)** TPR **b)** TNR **c)** Accuracy.

590   Figure 3 shows an assessment of techniques measuring accuracy, TPR,
591   and TNR in the Reuter datasets with mapper=3. The assessment of techniques
592   with the TPR measure is depicted in Figure 3a. For chunk size=3, the TPR
593   evaluated by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM,
594   BPLion+LFNN, and the proposed RIWO-based deep residual network were
595   0.747, 0.757, 0.771, 0.776, 0.780, 0.785, 0.790, and 0.803, respectively. Likewise,
596   for chunk size=6, the TPR evaluated using LSS-CNN, RNN,
597   SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, BPLion+LFNN, and the
598   proposed RIWO-based deep residual network were 0.800, 0.812, 0.818, 0.818,
599   0.818, 0.819, 0.819, and 0.830, respectively. The assessment of techniques
600   measuring TNR is depicted in Figure 3b. For chunk size=3, the TNR evaluated
601   using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN,
602   BPLion+LFNN, and the proposed RIWO-based deep residual network were
603   0.831, 0.842, 0.853, 0.861, 0.870, 0.880, 0.886, and 0.913, respectively. For chunk
604   size=6, the TNR evaluated using LSS-CNN, RNN, SLKNN+MLKNN, SVM,
605   NN, LSTM, BPLion+LFNN, BPLion+LFNN, and the proposed RIWO-based
606   deep residual network were 0.846, 0.850, 0.865, 0.869, 0.878, 0.885, 0.896, and
607   0.925, respectively. The assessment of the methods measuring accuracy is
608   depicted in Figure 3c. For chunk size=3, the accuracy evaluated using LSS-
609   CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN,
610   BPLion+LFNN, and the proposed RIWO-based deep residual network were
611   0.8306, 0.842, 0.852, 0.861, 0.870, 0.880, 0.886, and 0.913, respectively.
612   Likewise, for chunk size=6, the accuracy evaluated using LSS-CNN, RNN,
613   SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, BPLion+LFNN, and the
614   proposed RIWO-based deep residual network were 0.846, 0.850, 0.865, 0.869,
615   0.878, 0.885, 0.896, and 0.925, respectively. The performance improvement of
616   LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN,
617   BPLion+LFNN with respect to the proposed RIWO-based deep residual
618   network considering accuracy were 8.540%, 8.108%, 6.486%, 6.054%, 5.081%,
619   4.324%, and 3.135%, respectively.
620
621   b) Assessment with mapper=4
622

623   **Figure 4.** Assessment of different techniques comparing the proposed method and
624   the Reuters dataset with mapper=4 **a)** TPR **b)** TNR **c)** Accuracy.

625   We assessed the techniques by measuring accuracy, TPR, and TNR,
626   considering the Reuters dataset and using mapper=4 (Figure 4). The
627   assessment of techniques using TPR is displayed in Figure 4a. For chunk
628   size=3, the TPR evaluated using LSS-CNN was 0.754, RNN was 0.768,
629   SLKNN+MLKNN was 0.792, SVM was 0.796, NN was 0.800, LSTM was 0.806,
630   BPLion+LFNN was 0.810, and the proposed RIWO-based deep residual
631   network was 0.828. Likewise, for chunk size=6, the TPR evaluated using LSS-
632   CNN was 0.810, RNN was 0.820, SLKNN+MLKNN was 0.824, SVM was

0.824, NN was 0.825, LSTM was 0.826, BPLion+LFNN was 0.826, and the proposed RIWO-based deep residual network was 0.850. The assessment of techniques measuring TNR is depicted in Figure 4b. For chunk size=3, the TNR evaluated using LSS-CNN was 0.839, RNN was 0.860, SLKNN+MLKNN was 0.863, SVM was 0.870, NN was 0.875, LSTM was 0.881, BPLion+LFNN was 0.896, and the proposed RIWO-based deep residual network was 0.925. Likewise, for chunk size=6, the TNR evaluated by LSS-CNN was 0.855, RNN was 0.856, SLKNN+MLKNN was 0.876, SVM was 0.878, NN was 0.885, LSTM was 0.893, BPLion+LFNN was 0.900, and the proposed RIWO-based deep residual network was 0.940. The assessment of the methods measuring accuracy is displayed in Figure 4c. For chunk size=3, the accuracy evaluated by LSS-CNN was 0.837, RNN was 0.843, SLKNN+MLKNN was 0.846, SVM was 0.850, NN was 0.855, LSTM was 0.858, BPLion+LFNN was 0.862, and the proposed RIWO-based deep residual network was 0.880. For chunk size=6, the accuracy evaluated by LSS-CNN was 0.833, RNN was 0.849, SLKNN+MLKNN was 0.852, SVM was 0.857, NN was 0.859, LSTM was 0.863, BPLion+LFNN was 0.868, and the proposed RIWO-based deep residual network was 0.887. The performance improvement of LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN with respect to the proposed RIWO-based deep residual network and considering accuracy was 6.087%, 4.284%, 3.945%, 3.38%, 3.156%, 2.705%, and 2.142%, respectively.

4.5.2. Analysis with the 20 Newsgroups dataset

The assessment of techniques using the 20 Newsgroups datasets with TPR, TNR, and accuracy parameters was elaborated. The assessment was done with mapper=3 and mapper=4 and by altering the chunk size.

a) Assessment with mapper=3

Figure 5. Assessment of different techniques comparing the proposed method and the 20 Newsgroups dataset with mapper=3 **a)** TPR **b)** TNR **c)** Accuracy.

Figure 5 presents the assessment of techniques measuring accuracy, TPR and TNR and considering the 20 Newsgroups dataset with mapper=3. The assessment of techniques measuring TPR is depicted in Figure 5a. For chunk size=3, the maximum TPR of 0.834 was determined using the proposed RIWO-based deep residual network, while the TPR found using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.708, 0.759, 0.780, 0.785, 0.792, 0.803, and 0.812, respectively. Likewise, for chunk size=6, the highest TPR of 0.840 was found using the proposed RIWO-based deep residual network, while the TPR found using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.796, 0.815, 0.818, 0.822, 0.825, 0.828, and 0.829, respectively. The assessment of techniques measuring TNR is depicted in Figure 5b. For chunk size=3, the TNR computed using the proposed RIWO-based deep residual network was 0.862, while the TNR found using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.832, 0.839, 0.843, 0.846, 0.848, 0.850, and 0.851, respecitvely. Likewise, for chunk size=6, the TNR evaluated using the proposed RIWO-based deep residual network was 0.879, while those

computed by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.833, 0.840, 0.849, 0.808, 0.851, 0.852, and 0.854, respectively. The assessment of the methods measuring accuracy is depicted in Figure 5c. For chunk size=3, the accuracy evaluated by the proposed RIWO-based deep residual network was 0.850, while the accuracies computed by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion + LFNN were 0.821, 0.822, 0.833, 0.837, 0.839, 0.840, and 0.843, respectively. The performance improvement of LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion + LFNN with respect to the proposed RIWO-based deep residual network considering accuracy were 3.411%, 3.294%, 2%, 1.529%, 1.294%, 1.17%, and 0.823%, respectively. Likewise, for chunk size=6, the accuracy evaluated using the proposed RIWO-based deep residual network was 0.859, while the accuracies computed by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.824, 0.830, 0.839, 0.843, 0.849, 0.852, and 0.856, respectively.

b) Assessment with mapper=4

Figure 6. Assessment of different techniques compared with the proposed method considering the 20 Newsgroups dataset with mapper=4 **a)** TPR **b)** TNR **c)** Accuracy.

Figure 6 presents the assessment of techniques measuring accuracy, TPR, and TNR and considering the 20 Newsgroups dataset with mapper=4. The assessment of techniques with TPR measure is depicted in Figure 6a. For chunk size=3, the TPR evaluated by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual network were 0.721, 0.769, 0.798, 0.801, 0.803, 0.806, 0.810, and 0.845, respectively. Likewise, for chunk size=6, the TPR evaluated using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual network were 0.810, 0.827, 0.836, 0.849, 0.840, 0.843, 0.846, and 0.859, respectively. The assessment of techniques measuring TNR is depicted in Figure 6b. For chunk size=3, the TNR evaluated by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual networks were 0.831, 0.831, 0.842, 0.867, 0.848, 0.854, 0.859, and 0.870, respectively. Likewise, for chunk size=6, the TNR evaluated by LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual network were 0.836, 0.839, 0.851, 0.857, 0.863, 0.866, 0.871, and 0.910. The assessment of the method measuring accuracy is depicted in Figure 6c. For chunk size=3, the accuracy evaluated using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual network were 0.821, 0.831, 0.842, 0.846, 0.849, 0.854, 0.860, and 0.861, respectively. Likewise, for chunk size=6, the accuracy evaluated using LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN, and the proposed RIWO-based deep residual network were 0.824, 0.838, 0.849, 0.852, 0.858, 0.863, 0.868, and 0.870, respectively. The performance improvements of LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, BPLion+LFNN with respect to the proposed RIWO-based deep residual network and considering

729 accuracy were 5.287%, 3.678%, 2.413%, 2.068%, 1.379%, 0.804%, and 0.229%,
730 respectively.

731 *4.6. Comparative discussion*

732 Table 3 shows the assessment of techniques in terms of accuracy, TPR,
733 and TNR in the Reuters and 20 Newsgroups datasets. The Reuters dataset
734 with mapper=3 had a highest accuracy of 0.830 using the developed RIWO-
735 based deep residual network, while the accuracies of the existing LSS-CNN,
736 RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN were 0.800,
737 0.812, 0.818, 0.818, 0.818, 0.819, and 0.819, respectively. The proposed RIWO-
738 based deep residual network measured the maximum TPR of 0.925, while
739 LSS-CNN, RNN, SLKNN+MLKNN, SVM, NN, LSTM, and BPLion+LFNN
740 computed TPR of 0.846, 0.850, 0.865, 0.869, 0.878, 0.885, and 0.896,
741 respectively. The proposed RIWO-based deep residual network computed
742 the highest TNR of 0.880, while LSS-CNN, RNN, SLKNN+MLKNN, and
743 BPLion+LFNN computed TNR of 0.824, 0.839, 0.849, 0.852, 0.856, 0.859, and
744 0.863, respectively. With mapper=4, the highest TPR of 0.850, TNR of 0.940,
745 and accuracy of 0.887 were found using the developed RIWO-based deep
746 residual network. With the 20 Newsgroups datasets and mapper=3, the
747 highest TPR of 0.840, the highest TNR of 0.879, and the highest accuracy of
748 0.859 were computed by the proposed RIWO-based deep residual network.
749 With mapper=4, the highest TPR of 0.859, highest TNR of 0.910, and highest
750 accuracy of 0.870 were determined using the developed RIWO-based deep
751 residual network.

> **Comment [MT29]:** Please confirm if the changes made conform to the intended meaning of the sentence.
>
> **Comment [WU30]:** I agree with the change

752 **Table 3.** Comparative discussion.

753

754 **5. Conclusion**

755 This article presents a technique for text classification of big data
756 considering the MapReduce model. Its purpose is to provide a hybrid,
757 optimization-driven, deep learning model for text classification. Here, pre-
758 processing was carried out using stemming and stop word removal. The
759 mining of significant features was also performed wherein SentiWordNet,
760 contextual, and thematic features were mined from input pre-processed data.
761 Furthermore, the selection of the best features was carried out using the
762 Tanimoto similarity. The Tanimoto similarity examined the similarities
763 between the features and selected the pertinent features with higher feature
764 selection accuracy. Then, a deep residual network was employed for dynamic
765 text classification. The Adam algorithm trained the deep residual network
766 and dynamic learning was carried out with the proposed RIWO-based deep
767 residual network and fuzzy theory for incremental text classification. Deep
768 residual network training was performed using the proposed RIWO. The
769 proposed RIWO algorithm is the integration of IWO and ROA, and it
770 outperformed other techniques with the highest TPR of 85%, TNR of 94%,
771 and accuracy of 88.7%. The proposed method's performance will be
772 evaluated using different datasets in the future. Additionally, bias mitigation
773 strategies that do not directly depend on a set of identity terms and methods
774 that are less dependent on individual words will be considered to effectively

775   deal with biases tied to words used across many different contexts, e.g., white
776   vs. black.
777

Comment [MT31]: Please confirm the changes conform to the intended meaning of the sentence.

Comment [WU32]: I agree the change

778   **References**

779   1.   Ranjan, N.M. and Prasad, R.S.," LFNN: Lion fuzzy neural network-based evolutionary model for text
780   classification using context and sense based features", Applied Soft Computing, vol. 71, pp.994-1008, 2018.
781   2.   Kotte, V.K., et al., "A similarity function for feature pattern clustering and high dimensional text document
782   classification", Foundations of Science, pp.1-18, 2019.
783   3.   Wang, Jiaying; et al., "Large-Scale Text Classification Using Scope-Based Convolutional Neural Network: A
784   Deep Learning Approach", IEEE Access, vol.7, pp.171548–171558, 2019.
785   4.   Kuppili, Venkatanareshbabu; et al., A Mechanics-Based Similarity Measure for Text Classification in Machine
786   Learning Paradigm", IEEE Transactions on Emerging Topics in Computational Intelligence, pp. 1–21, 2018.
787   5.   Liu, C. and Wang, X., "Quality-related English text classification based on recurrent neural network", Journal
788   of Visual Communication and Image Representation, vol.71, pp.102724, 2020.
789   6.   Qi Li, et al.,"Improving convolutional neural network for text classification by recursive data pruning",
790   Neurocomputing, vol. 414, pp. 143–152, 2020.
791   7.   BenSaid, F. and Alimi, A.M., "Online feature selection system for big data classification based on multi-
792   objective automated negotiation", Pattern Recognition, vol. 110, pp.107629, 2020.
793   8.   Jiang, M., et al., "Text classification based on deep belief network and softmax regression", Neural
794   Computing and Applications, vol. 29, no. 1, pp.61-70, 2018.
795   9.   T. W. Schoenharl and G. Madey, "Evaluation of Measurement Techniques for the Validation of Agent-Based
796   Simulations Against Streaming Data",Berlin, Germany: Springer, 2008.
797   10.   C. G. Gonzalez, et al., "Density of ´ closed balls in real-valued and autometrizedboolean spaces for clustering
798   applications," In Proceedings of 19th Brazilian Symposium of Artificial Intelligence, pp. 8–22, 2008.
799   11.   Kingma, D.P. and Ba, J., "Adam: A method for stochastic optimization", 2014.
800   12.   S. Tata and J. M. Patel, "Estimating the selectivity of tf-idf based cosinesimilarity predicates,"ACMSigmod
801   Rec., vol. 36, no. 2, pp. 7–12, 2007.
802   13.   SiweiLai,et al.,''Recurrent convolutional neural networks for text classification'', In Proceedings of the
803   Twenty-Ninth AAAI Conference on Artificial Intelligence,pp. 2267–2273,2015.
804   14.   T. Wu, et al., ''Twitter spam detection based ondeep learning,'' in Proc. ACSW, Art. no. 3, 2017.
805   15.   M. V. Mäntylä, et al., ''The evolution of sentiment analysis—A review of research topics, venues, and top
806   cited papers'', Computer Science Review, vol. 27, pp. 16–32, Feb. 2018.
807   16.   Junxian He, et al., "Efficient correlated topic modeling with topic embedding", In Proceedings of the 23rd
808   ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 225-233, 2017.
809   17.   Yonghui Wu, et al., "Google's neural machine translation system: Bridging the gap between human and
810   machine translation", 2016.
811   18.   Kim, J.K., et al.,"Intent detection using semantically enriched word embeddings",In 2016 IEEE Spoken
812   Language Technology Workshop (SLT), pp. 414-419,December   2016.
813   19.   Chen, J., et al.,"Verbal aggression detection on Twitter comments: Convolutional neural network for short-
814   text sentiment analysis", Neural Computing and Applications, pp.1-10, 2018.
815   20.   Cheng, P. M., andMalhi, H. S.,"Transfer learning with convolutional neural networks for classification of
816   abdominal ultrasound images", Journal of Digital Imaging, vol. 30, no. 2, pp. 234-243, 2017.
817   21.   Esteva, A., et al.,"Dermatologist-level classification of skin cancer with deep neural networks", Nature, vol.
818   542, no. 7639, pp. 115-118, 2017.
819   22.   Luo, and Yuan., "Recurrent neural networks for classifying relations in clinical notes", Journal of
820   BiomedicalInformatics,vol. 72, pp85-95, 2017.
821   23.   Zhang, M., et al.,"Collaborative classification of hyperspectral and visible images with convolutional neural
822   network", Journal of Applied Remote Sensing,vol. 11, no. 4, pp. 1, 2017.
823   24.   20newgroup dataset extracted from, "https://www.kaggle.com/crawford/20-newsgroups," accessed on
824   November 2020.
825   25.   Reuter dataset taken from, " https://www.kaggle.com/nltkdata/reuters," accessed on November 2020.
826   26.   Chen, Z., et al.,"Deep residual network-based fault detection and diagnosis of photovoltaic arrays using
827   current-voltage curves and ambient conditions", Energy Conversion and Management, vol. 198, pp.111793,
828   2019.

829   27. Sang, H.Y., et al.,"An effective invasive weed optimization algorithm for scheduling semiconductor final
830   testing problem", Swarm and Evolutionary Computation, vol. 38, pp.42-53, 2018.
831   28. Binu, D. and Kariyappa, B.S., "RideNN: A new rider optimization algorithm-based neural network for fault
832   diagnosis in analog circuits" IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 1, pp.2-26,
833   2018.
834   29. DHarsha Dave, and Shree jaswal; "Multiple text document summarization system using hybrid
835   summarization technique", In IEEE 1st International Conference on Next Generation Computing
836   Technologies (NGCT), pp. 804-808, September 2015.
837   30. OguzhanTas, and FarzadKiyani; "A survey automatic text summarization", PressAcademia Procedia, vol.5,
838   no.1, pp.205-213, 2007.
839   31. Ghosh, M. and Kar, A., "Unsupervised linguistic approach for sentiment classification from online reviews
840   using SentiWordNet 3.0," Int J Eng Res Technol, vol.2, no.9, 2013.
841   32. VidyadhariCh,et al., "A Semantic Word Processing Using Enhanced Cat Swarm Optimization Algorithm for
842   Automatic Text Clustering", Multimedia Research, Vol.2,No.4, pp.23-32,2019.
843   33. Abdalla, H. B., et al.,   (2020). Optimization Driven MapReduce Framework for Indexing and Retrieval of Big
844   Data. KSII Transactions on Internet and Information Systems (TIIS), 14(5), 1886-1908.
845   34. M. Mohsin, H. Li and H. B. Abdalla, "Optimization Driven Adam-Cuckoo Search-Based Deep Belief Network
846   Classifier for Data Classification," in IEEE Access, vol. 8, pp. 105542-105560, 2020, doi:
847   10.1109/ACCESS.2020.2999865.
848   35. Farman Ali, et al.,"An intelligent healthcare monitoring framework using wearable sensors and social
849   networking data," Future Generation Computer Systems, vol.114, pp.23-43, 2021.
850   36. Farman Ali, et al., "Traffic accident detection and condition analysis based on social networking data,"
851   National library of medicine, 2021.
852   37. Farman Ali, et al., "Fuzzy Ontology and LSTM-Based Text Mining: A Transportation Network Monitoring
853   System for Assisting Travel," Sensors, vol.19, no.2, pp.234, 2019.
854   38. Farman Ali, et al.,Department of Information and Communication Engineering, Inha University, Incheon,
855   South Korea, "Merged Ontology and SVM-Based Information Extraction and Recommendation System for
856   Social Robots," IEEE Xplore, vol.5, pp.12364-12379, 2017.
857   39. Muhammad Pervez Akhter, Zheng Jiangbin, Irfan Raza Naqvi, Mohammed Abdelmajeed, Atif Mehmood,
858   and Muhammad Tariq Sadiq, "Document-Level Text Classification Using Single-Layer Multisize Filters
859   Convolutional Neural Network," IEEE Access, vol. 8, pp.42689 - 42707, 2020.
860   40. Christopher A. Flores, Rosa L. Figueroa, and Jorge E. Pezoa, "Active Learning for Biomedical Text
861   Classification Based on Automatically Generated Regular Expressions," IEEE Access, vol. 9, pp.38767 - 38777,
862   2021.
863   41. Hai Huan, Jiayu Yan, Yaqin Xie, Yifei Chen, Pengcheng Li, and Rongrong Zhu, "Feature-Enhanced
864   Nonequilibrium Bidirectional Long Short-Term Memory Model for Chinese Text Classification,"   IEEE
865   Access, vol. 8, pp. 199629 - 199637, 2020.
866   42. Yanru Dong, Peiyu Liu, Zhenfang Zhu, Qicai Wang, and Qiuyue Zhang, "A Fusion Model-Based Label
867   Embedding and Self-Interaction Attention for Text Classification," vol.8, pp. 30548 - 30559, 2019.

Figure 1

Schematic view of text classification from the input big data using proposed RIWO-based Deep Residual Network

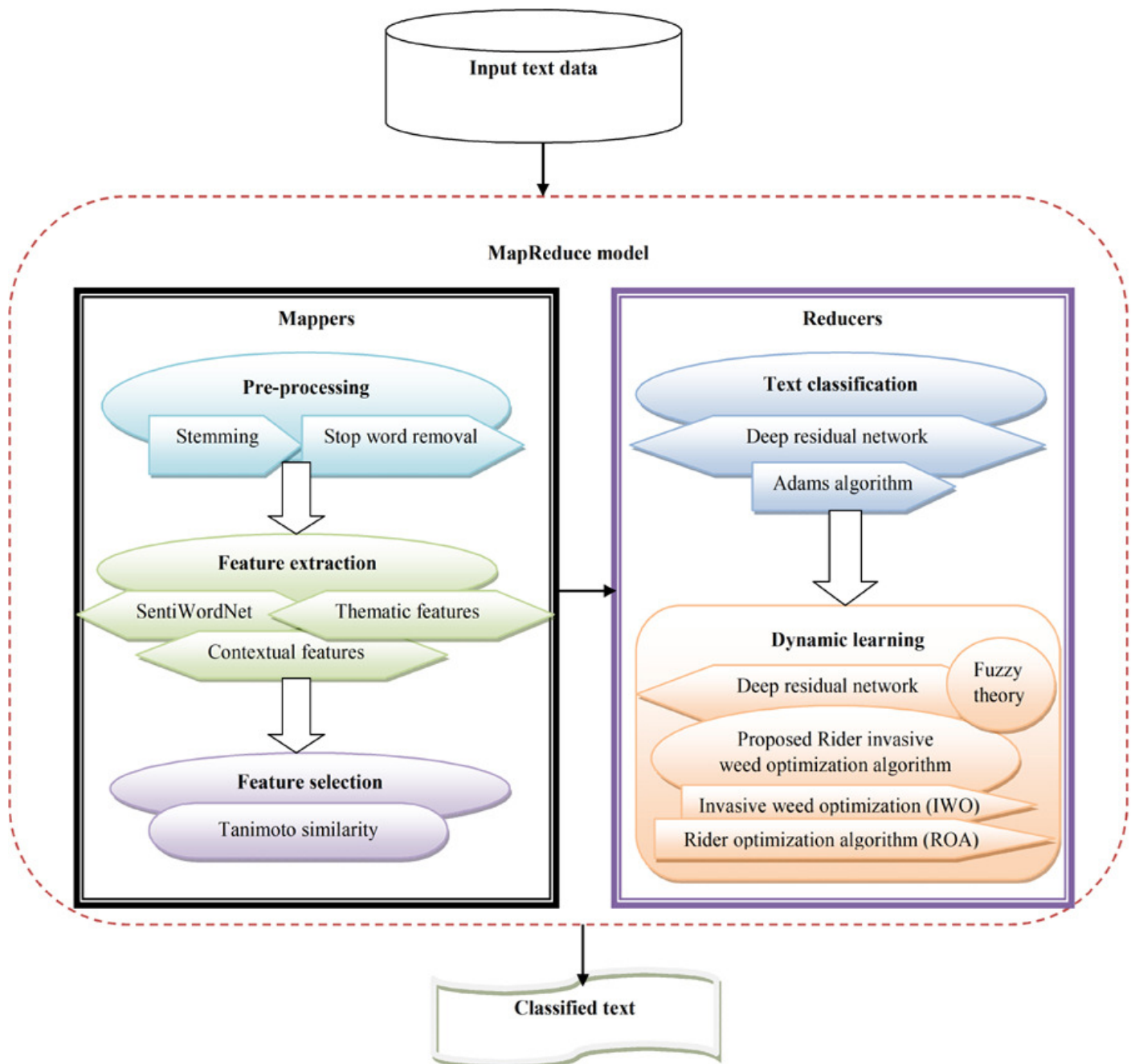



Figure 2

Structural design of deep residual network with residual blocks, convolutional (Conv) layers, linear classifier, and average pooling layers for text classification

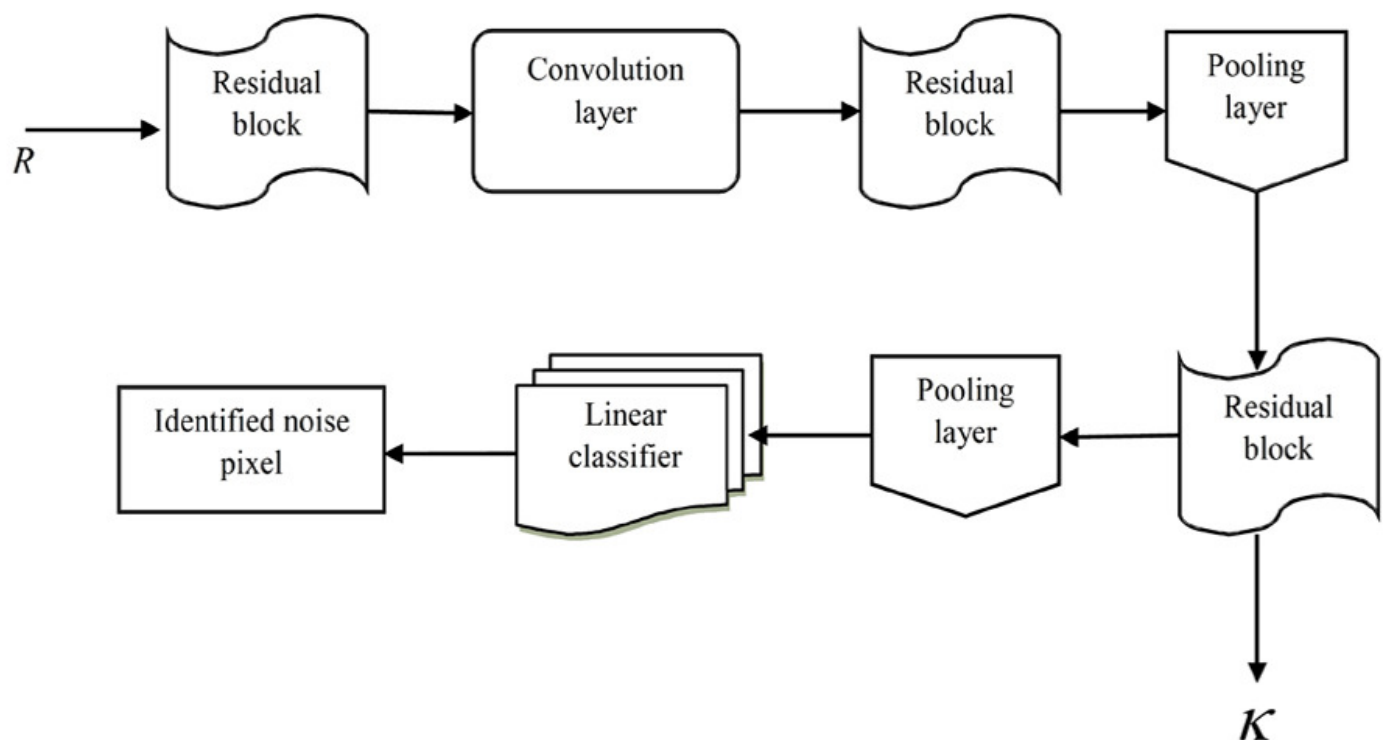



Figure 3

Assessment of different techniques comparing with the proposed method by considering Reuter dataset with mapper=3 a) TPR b) TNR c) Accuracy

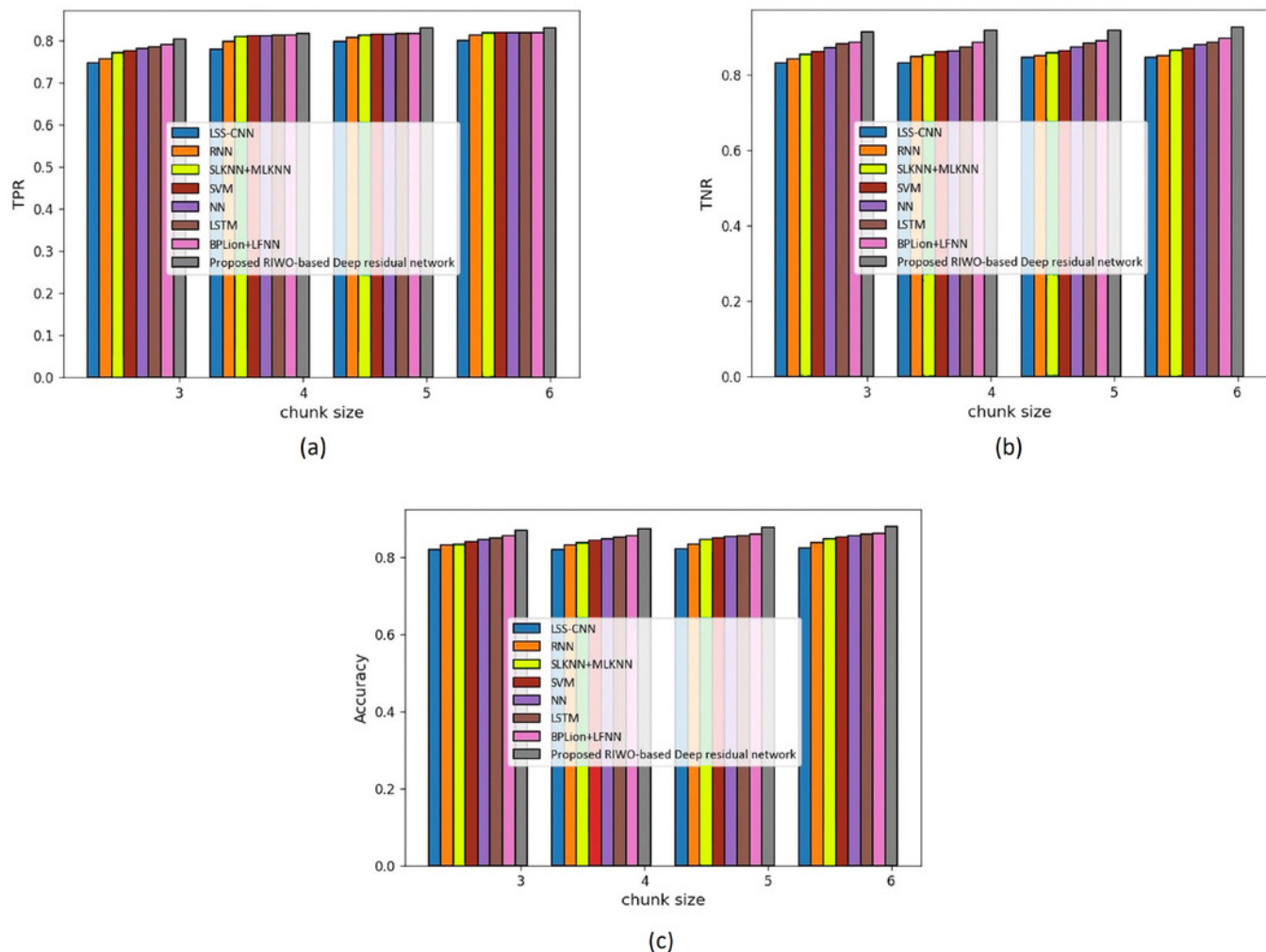



Figure 4

Assessment of different techniques comparing with the proposed method by considering Reuter dataset with mapper=4 a) TPR b) TNR c) Accuracy

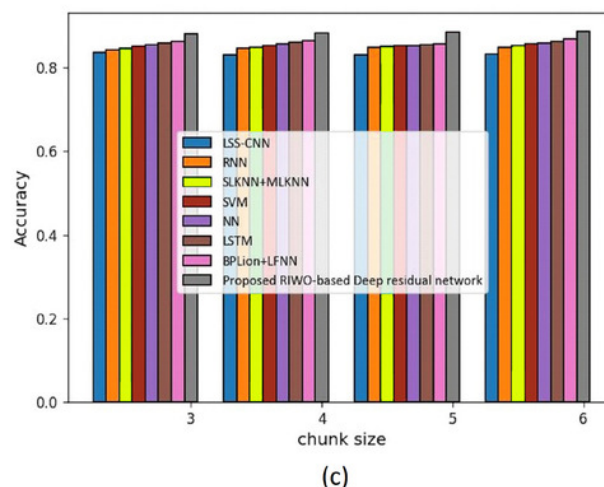
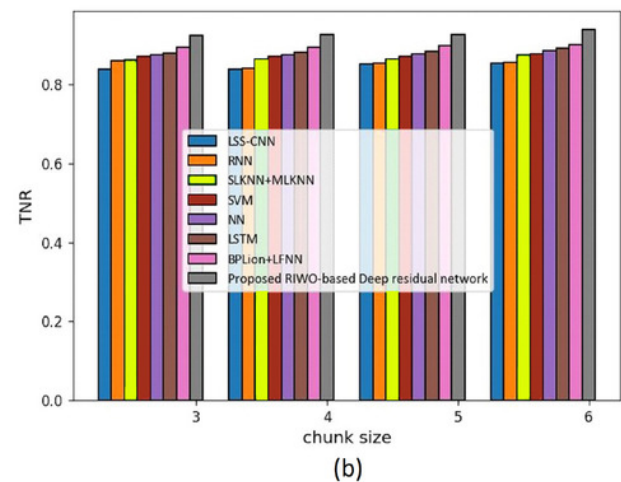
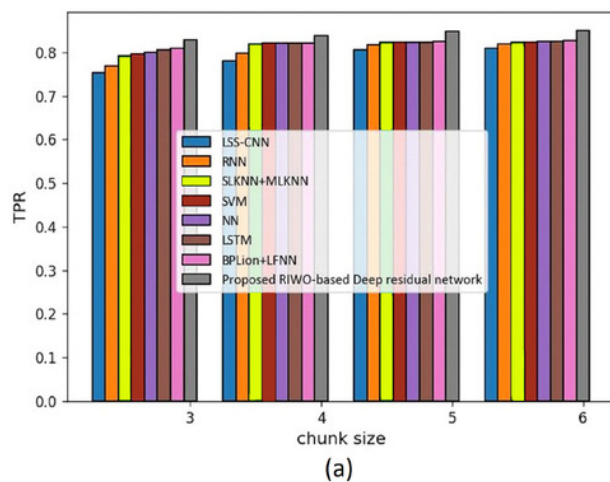



Figure 5

Assessment of different techniques comparing with the proposed method by considering 20 Newsgroup dataset with mapper=3 a) TPR b) TNR c) Accuracy

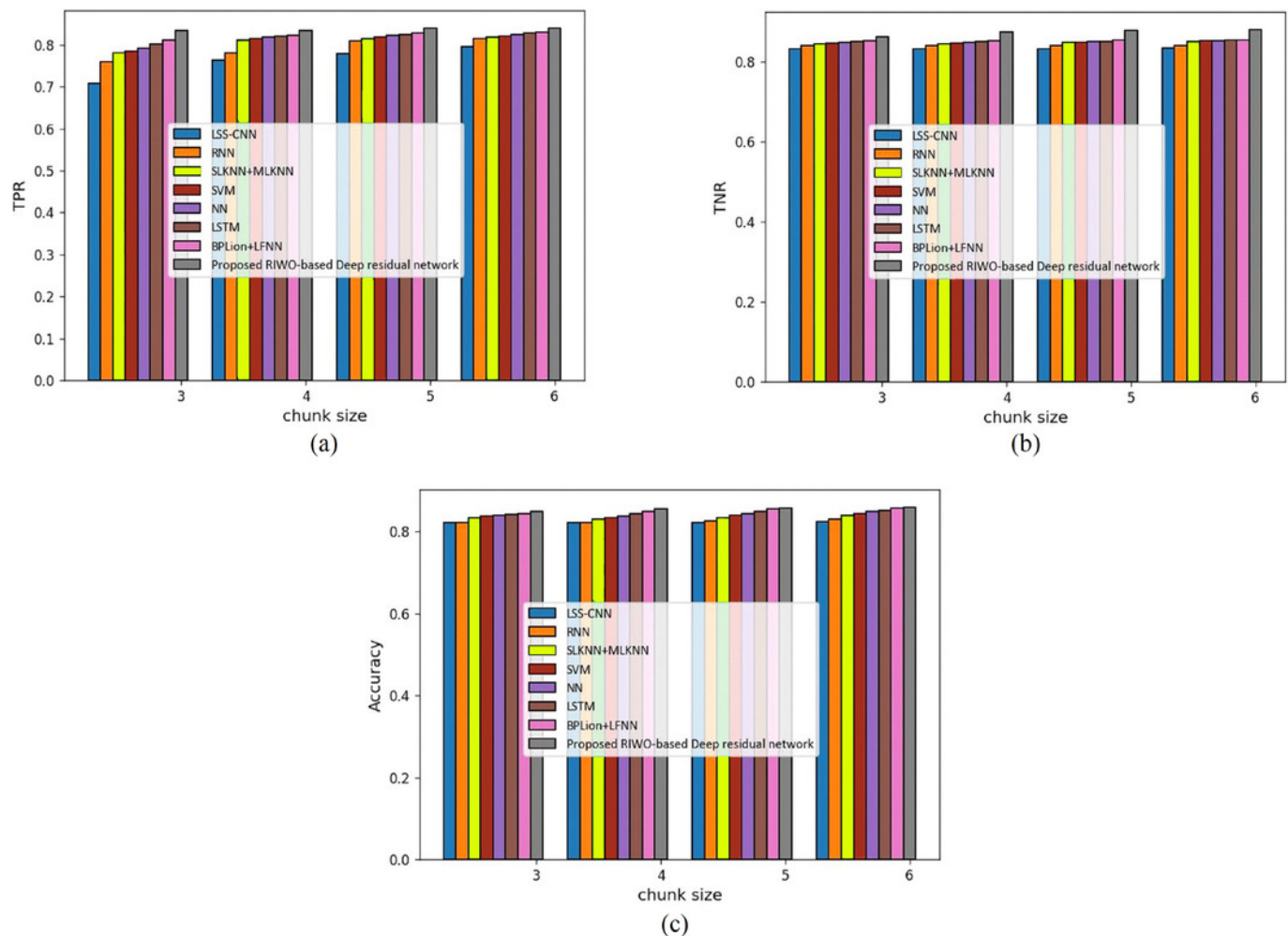
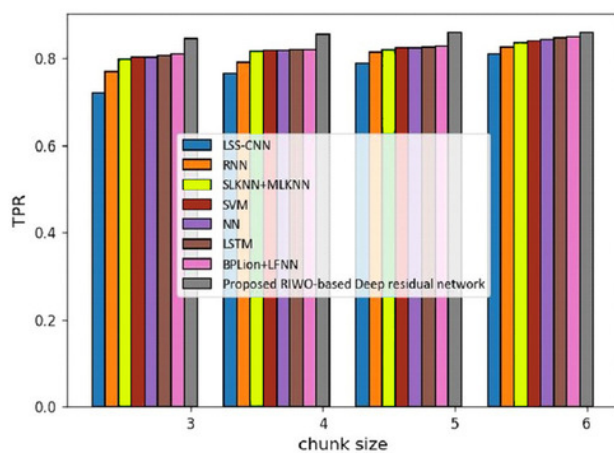
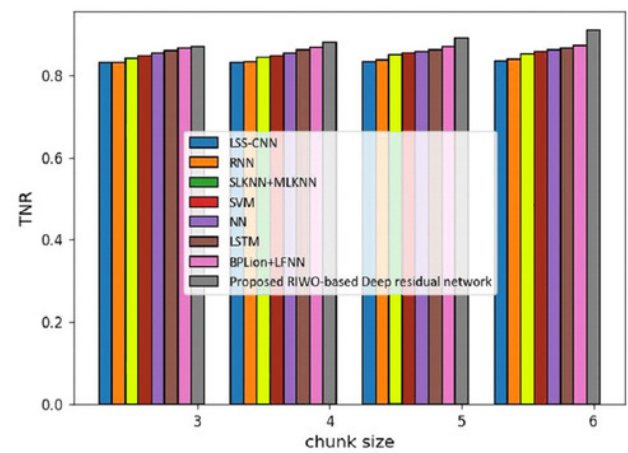
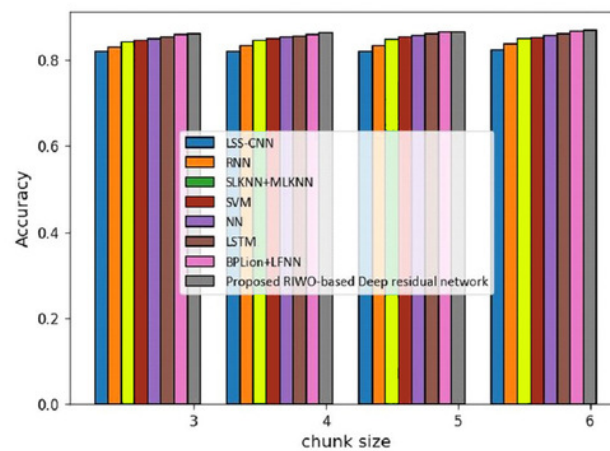



Figure 6

Assessment of different techniques comparing with the proposed method by considering 20 Newsgroup dataset with mapper=4 a) TPR b) TNR c) Accuracy

Pseudocode of Adams algorithm

Input: $\hat{q}_l$: corrected bias of first moment estimate and $\hat{m}_l$: corrected bias of second moment estimate

---

Output: Resulting parameters

Require α step size

Require $\eta_1, \eta_2 \in [0,1]$: Exponential rate decay

Initialize first-moment vector

Initialize second-moment vector

Initialize time step

While θ_l not converged do

$l \leftarrow l + 1$

Get gradients

Update first biased moment estimate using equation (22)

Update second biased moment estimate using equation (24)

Evaluate corrected bias first-moment estimate

Evaluate corrected bias second-moment estimate

Update parameters

End while

Return θ_l

Pseudocode of developed RIWO

Input: A : Arbitrary rider position, n : iteration, n_{max} : maximum iteration

---

Output: Leader A^L

Begin

Initialize solutions set

Initialize algorithmic parameter

Discover error using equation (19)

While $n < N_{OFF}$

For $v = 1\ to\ P$

Update bypass position with equation (29)

Update follower position with equation (39)

Update overtaker position with equation (27)

Update attacker position with equation (28)

Rank riders using error with equation (19)

Choose the rider with minimal error

Update steering angle, gear, accelerator, and brake

Return A^L

$n = n + 1$

End for

End while

End

Comparative discussion

Datasets	Mappers	Metrics	LSS-CNN	RNN	SLKNN + MLKNN	SVM	NN	LSTM	BPLion+ LFNN	Proposed RIWO-based deep residual network
Reuters dataset	Mapper =3	TPR	0.800	0.812	0.818	0.818	0.818	0.819	0.819	**0.830**
		TNR	0.846	0.850	0.865	0.869	0.878	0.885	0.896	**0.925**
		Accuracy	0.824	0.839	0.849	0.852	0.856	0.859	0.863	**0.880**
	Mapper =4	TPR	0.810	0.820	0.824	0.824	0.825	0.826	0.826	**0.850**
		TNR	0.855	0.856	0.876	0.878	0.885	0.893	0.900	**0.940**
		Accuracy	0.833	0.849	0.852	0.857	0.859	0.863	0.868	**0.887**
20 Newsgroup dataset	Mapper =3	TPR	0.796	0.815	0.818	0.822	0.825	0.828	0.829	**0.840**
		TNR	0.833	0.840	0.849	0.851	0.855	0.852	0.854	**0.879**
		Accuracy	0.824	0.830	0.839	0.843	0.849	0.852	0.856	**0.859**
	Mapper =4	TPR	0.810	0.827	0.836	0.840	0.843	0.846	0.849	**0.859**
		TNR	0.836	0.839	0.851	0.857	0.863	0.866	0.871	**0.910**
		Accuracy	0.824	0.838	0.849	0.852	0.858	0.863	0.868	**0.870**