Towards Dimension Reduction: A Balanced Relative Discrimination Feature Ranking Technique for Efficient Text Classification (BRDC)

Muhammad Nasir[1], Noor Azah Samsudin[2], Wareesa Sharif[3],
Souad Baowidan[4]*, Dr. Humaira Arshad[5], Muhammad Faheem Mushtaq[6]
Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn (UTHM), Johor Bahru, Malaysia[1]
Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn (UTHM), Johor Bahru, Malaysia[2]
Department of Artificial Intelligence, The Islamia University of Bahawalpur (The IUB), Bahawalpur, Pakistan[3]
Faculty of Computing and IT, King Abdulaziz University, Jeddah, Saudi Arabia[4]
Department of Computer Science, The Islamia University of Bahawalpur (The IUB), Bahawalpur, Pakistan[5]
Department of Information Technology, The Islamia University of Bahawalpur (The IUB), Bahawalpur, Pakistan[6]

***Abstract*—The volume and complexity of textual data have significantly increased worldwide, demanding a comprehensive understanding of machine learning techniques for accurate text classification in various applications. In recent years, there has been significant growth in natural language processing (NLP) and neural networks (NNs). Deep learning (DL) models have outperformed classical machine learning approaches in text classification tasks, such as sentiment analysis, news categorization, question answering, and natural language inference. Dimension reduction is crucial for refining the classifier performance and decreasing the computational cost of text classification. Existing methodologies, such as the Improved Relative Discrimination Criterion (IRDC) and the Relative Discrimination Criterion (RDC), exhibit deficiencies in proper normalization and are not well-balanced regarding distinct class's term ranking. This study introduced an improved feature-ranking metric called the Balanced Relative Discrimination Criterion (BRDC). This study measured document frequencies into term-count estimations, facilitating a normalized and balanced classification approach. The proposed methodology demonstrated superior performance compared to existing techniques. Experiments were conducted to evaluate the efficacy of the proposed techniques using Decision Tree (DT), Logistic Regression (LR), Multinomial Naïve Bayes (MNB), and Long Short-Term Memory (LSTM) models on three benchmark datasets: Reuters-21578, 20newsgroup, and AG News. The findings indicate that LSTM outperformed the other models and can be applied in conjunction with the proposed BRDC approach.**

Keywords*—Text classification; balanced relative discrimination criterion; dimension reduction; feature ranking; deep learning; machine learning

I. Introduction

Owing to the persistent expansion of information technology, the production of available information poses a substantial challenge, and to manage big data has garnered considerable attention. Approximately 80% of companies manage and arrange their data in a text format. [1, 2], and is increasing daily. Classification is dynamic in machine learning, particularly text classification, in which text documents are automatically sorted into predefined categories. Various machine learning classifiers, such as Decision Trees (DT), Logistic Regression (LR), and Multinomial Naïve Bayes (MNB), have been used to evaluate text classification performance [3, 4]. Text classification is a fundamental approach for detecting and classifying textual data [5].

The performance of a model is influenced by several factors, with input data being one of the most important. Various types of textual datasets were used to increase the number of input variables in the model. However, the high dimensionality of the feature space can hinder the text classification performance. Thus, reducing dimensionality is a critical challenge in text classification [6]. Not all features hold equal significance in datasets comprising text with high-dimensional features, and some may be redundant, irrelevant, or noisy. To classify a document into different classes, the discriminative capabilities of features are used in machine learning algorithms to solve a given classification problem, where each feature is represented as a discrete characteristic [7]. This helps reduce the computational cost and increases the performance and prediction accuracy [8]. As document collections increase, there is a need for more advanced information processing methods to search, retrieve, and organize text efficiently. Machine learning approaches have accomplished superior performance, resulting in natural language handling. The outcome of these learning approaches depends on their ability to grasp complex models and non-straight connections within information. Nonetheless, tracking reasonable designs, models, and methods for text characterization is difficult for specialists [9]. Removing redundant and irrelevant variables from the input data before proceeding with the model is crucial. This is performed using feature selection, which decreases the computational cost and improves prediction accuracy by providing enhanced and minimized data [8]. Feature selection is essential when using high-dimensional datasets in which the number of observations is less than the number of features [10]. The significant contributions of this study are as follows:

- Proposed BRDC for Text Classification that increases classification.

- Purpose a normalized and balanced technique, compared to RDC and IRDC, for a balanced and efficient text classification.
- Reduce the number of iterations to calculate the AUC.
- A definite integral-based method to calculate the Area Under the Curve (AUC).
- The classification experiments used both machine learning and deep learning models.
- The proposed model was compared using three balanced and unbalanced datasets.

This research proposes a normalized term ranking approach in which each term in distinct classes gets a balanced rank. The proposed feature ranking approach, Balanced Relative Discriminant Criterion (BRDC), compared with existing feature ranking approaches such as Relative Discriminant Criterion (RDC) and Improved Relative Discriminant Criterion (IRDC), experiments results show the proposed approach outperforms the in comparison to the existing approaches.

II. Literature Review

Text classification involves categorizing large volumes of text into one or more predefined categories based on the content or characteristics of the text [11]. In this section, we explain the different feature ranking techniques used for various types of classification, most of which are based on document frequency. Feature ranking techniques can be categorized into three types: filter-based, wrapper, and embedded [12, 13]. The leading causes of declining algorithmic performance in text classification are categorization and feature extraction from documents that employ the extracted features. The primary purpose of the feature ranking technique is to reduce the dimensionality of the dataset(s) by eliminating irrelevant features for classification. Dimension reduction has several advantages, such as reducing the dataset size, lowering the computational demands of text categorization algorithms (particularly those that do not scale well with large feature sets), and significantly reducing the search space [14]. A study demonstrated how applying bagging and Bayesian boosting techniques to classification algorithms, such as Multinomial Naïve Bayes (MNB) and K-nearest neighbor (K-NN), can improve their performance [15]. To determine which strategy was most effective in capturing text features and enabling the classifier to achieve the highest accuracy, a study analyzed the outcomes of applying three text feature extraction algorithms while classifying short sentences and phrases using a neural network. Term frequency Inverse Document Frequency (TF-IDF) and its two variations, which use various dimensionality reduction approaches, are among the feature extraction methods explored. A document frequency-based comparison was performed using Term Frequency Inverse Document Frequency (TF-IDF), Latent Semantic Analysis (LSA), and Linear Discriminant Analysis (LDA), and the results showed that the document frequency-based technique performed well [16].

There are two main methods for minimizing the dimensions of the feature vectors. Feature selection is the first approach to creating a new subset of the initial feature collection. Feature extraction is the second method for reducing dimensions. It makes a new feature set in a new feature space with smaller dimensions. The linear separability of the classes determines whether the two techniques are linear or nonlinear [17]. One study assessed and analyzed three Stemming methods. They are Light-Stemming Root-Based-Stemming, and Dictionary-Based Stemming. The intention is to decrease the element space into an information space with a much lower aspect ratio for two cutting-edge classifiers: artificial neural networks and support vector machines (SVM) [18]. Document Frequency (DF) and Term Variance (TV)-based methods were proposed for feature selection, and the next Principal Component Analysis (PCA) method was applied to reduce further the features, which were tested on the Reuters-21578 benchmark dataset and showed effective results [19]. The filter-based technique is typically faster and independent of the induction algorithm's function, meaning the selected feature can be input to any model's algorithm for further processing [20]. To identify a reliable strategy that can be applied to real datasets, one study evaluated the effectiveness of several feature selection techniques under diverse scenarios using synthetic datasets, in which different filtering measures can be employed for classification, such as distance, dependence, information, statistical measures, and consistency [21, 22], such as chi-square and information gain [23, 24]. A study evaluated machine learning methods for serial analysis of gene expression (SAGE)--based cancer classification, suggesting using chi-square for gene selection to address the high dimensionality in the dataset. The support vector machine (SVM) and Naive Bayes (NB) emerged as top-performing classifiers, and chi-square selection improved the performance across all methods. These experiments were conducted on human brain and breast SAGE datasets. It uses the principal criteria for variable selection by ordering the filter technique using the variable ranking method. Filter-based techniques are frequently used because of their simplicity and exemplary performance in real-life applications. This technique uses a threshold as a suitable rambling criterion to score a variable [25]. When we talk about real-world applications, owing to the heavy reliance on clustering, the wrapper-based technique is unsuitable mainly because it requires clusters, and to evaluate clustering in diverse subspaces, there is a lack of suitable clustering criteria [26].

A feature ranking metric named relative discrimination criterion (RDC) [27] considers both document frequencies and term count to estimate the importance of a term; in this study, the performance of RDC is compared using two classifiers such as Support Vector Machine (SVM) and Naive Bayes (NB) classifiers on benchmark datasets, the said technique is not well normalized. However, the RDC technique needs to be normalized, and an optimal and balanced solution for dimension reduction is required. Another feature ranking technique was introduced and named the Improved Relative Discriminative Criterion (IRDC) [28], which uses document and term frequencies to rank terms. IRDC prioritizes rarely occurring terms over frequently occurring ones. IRDC focuses on rarely occurring terms present in one class and absent in others, thereby achieving a balance between frequent and rare terms. The experimental results in this study show that IRDC outperforms existing techniques in terms of the F-measure on datasets such as Reuters-21578 and 20newsgroup using classifiers such as Decision Tree (DT), Naïve Based (NB), and Support Vector

Machine (SVM), which also need to optimize the data to achieve a better result.

A study introduced a novel approach called the De-redundancy Relative Discrimination Criterion (DRDC), designed to assess terms' importance while considering their redundancy [29]. DRDC incorporates the Relative Discrimination Criterion (RDC) and Mutual Information (MI) to gauge term relevance to categories and the redundancy between terms. During the selection process, the RDC and mutual information scores were normalized separately to balance them and mitigate the impact of mutual information. A study merged the Relative Discrimination Criterion (RDC) with Ant Colony Optimization (ACO) in a two-stage feature selection (FS) technique [30]. Initially, the RDC ranks the features based on their values, and those with values lower than a threshold are eliminated from the feature set. Subsequently, the ACO-based feature selection method acts as a wrapper method for selecting redundant or irrelevant features that are not eliminated in the first stage. The experimental results demonstrate the efficacy of the RDC-ACO method for text feature selection.

III. Proposed Approach

This study proposes a Balanced Relative Discrimination Criterion (BRDC) that uses normalization and balanced approaches for feature ranking to increase the accuracy and performance of the model. This study consisted of four main stages, explained in detail in this section. The proposed technique calculates the document of each term count to obtain information from the given text. The BRDC considers the differences between the DF and the respective Term Counts (TC) in the positive and negative classes. In previous studies, an effective measure using DF has been used for feature selection in textual data classification. It calculates the number of documents and their terms in a specific class and counts them as a feature, which can be a specially derived attribute, word, or sentence. If a document contains a feature, the DF increases by 1. Traditional DF metric counting has a drawback because it does not consider the importance of a feature in a specific document [31]; therefore, the term count is ignored when ranking a particular term [32]. In this proposed approach, terms count ranked in distinct classes in a balanced way. Two standard techniques are used to build a multiclass classifier, namely one-against-one and against-all, to break down multiclass classification problems into binary classification problems [33]. This means the multiclass problem is usually divided into multiple two-class issues, where one class is positive, and all other courses are combined to form a negative class. The dataset comprised documents categorized into classes designed for training and evaluating algorithms on new documents. Three single-labeled datasets of varying sizes and class distributions were used: Reuter-21578, 20newsgroup, and AG News. These datasets are considered the standard for text classification and were sourced from The UCI Machine Learning Repository. Previous studies have widely used these methods [34-37]. Fig. 1 shows the overall working flow of the proposed model, which consists of four steps. BRDC is tailored for text classification and comprises four stages: preprocessing, feature selection, data modeling, and the last state as a post-analysis. The raw text underwent several preprocessing steps, such as tokenization, stemming, and stop-word removal. One class is treated as the positive class to handle binary to multi-class classification. In contrast, all other classes are combined to form the negative class used in this study for classification. Fig. 1 shows the overall classification step.

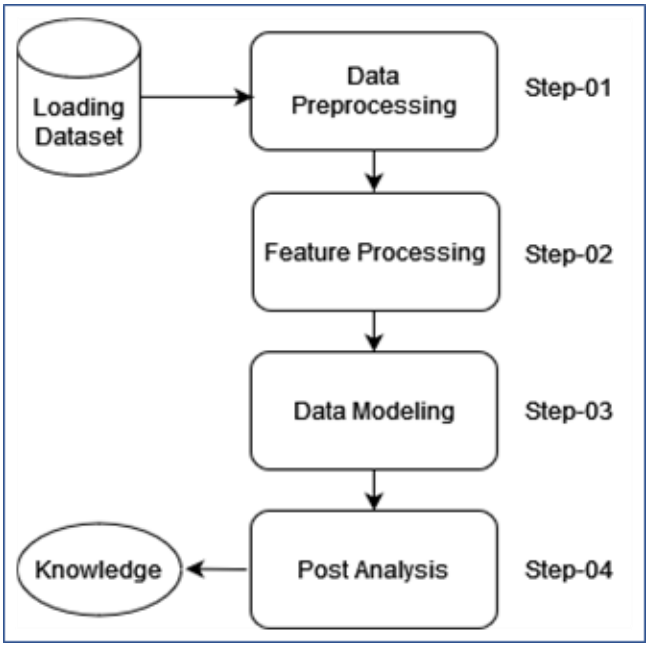

Fig. 1. Overall framework of BRDC.

TABLE I. List of Documents in each Negative and Positive Class

Document	Class	Content of Document
Doc1	Positive	Red, Blue, Green, Green, Yellow
Doc2	Positive	Red, Green, Blue, Red. Yellow
Doc3	Positive	Green, Blue, Red, Yellow, Yellow
Doc4	Positive	Green, Red, Blue, Green, Blue
Doc5	Positive	Blue, Red, Blue, Red, Yellow, Green
Doc6	Positive	Blue, Green, Yellow, Green
Doc7	Positive	Yellow, Green, Blue, Red
Doc8	Positive	Yellow, Yellow, Red, Green, Red
Doc9	Negative	Blue, Green, Blue
Doc10	Negative	Green, Green, Yellow
Doc11	Negative	Red, Green, Red
Doc12	Negative	Green, Blue, Yellow
Doc13	Negative	Blue, Red
Doc14	Negative	Green, Blue, Green,
Doc15	Negative	Blue, Red, Red
Doc16	Negative	Green, Yellow, Yellow

Table I shows the total number of documents, the class of the document, and the content of the document.

Fig. 2 shows the process of converting the document into terms, which consists of a document “Deep learning is a subset of machine learning,” which contains the following terms: Deep count is 1, learning 2 is 1, a 1 subset 1 of 1, and machine 1.

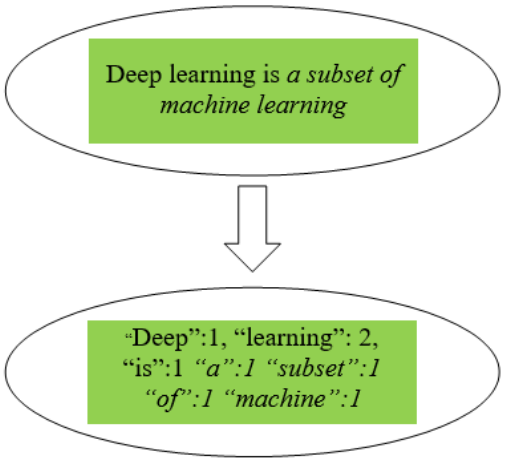

Fig. 2. ELABORATE the distinct word.

A. Feature Ranking Metric

Text classification feature selection metrics are typically based on a word's term or document frequency [38]. Most feature-ranking techniques use document frequency, such as chi-square, to calculate the term rank of features in a textual dataset [39]. Categorical document frequency indicates the dispersal of a term over a separate category [40]. To determine the required rank using term counts, the document frequency is divided into the average of all frequencies for each term count, and this concept is considered a sample dataset. A balanced dataset significantly improves the data mining process [41].

Text normalization is critical in language- and speech-based application tasks [42]. This study also focused on proposing a Balanced Relative Discrimination Criterion (BRDC) feature ranking technique for text classification with a more normalized discriminant method, which normalized each term count by dividing the average values of all term counts.

Table II consists of sixteen documents with four different terms; from one to eight, there are positive class documents, and from nine to sixteen, there are negative documents.

TABLE II. TERM COUNT FOR EACH TERM ACCORDING TO THEIR CLASS

Document	Class	F1	F2	F3	F4
Doc1	Positive	1	1	2	1
Doc2	Positive	2	1	1	1
Doc3	Positive	1	1	1	2
Doc4	Positive	1	2	2	0
Doc5	Positive	2	2	1	1
Doc6	Positive	0	1	2	1
Doc7	Positive	1	1	1	1
Doc8	Positive	2	0	1	2
Doc9	Negative	0	2	1	0
Doc10	Negative	0	0	2	1
Doc11	Negative	2	0	1	0
Doc12	Negative	0	1	1	1
Doc13	Negative	1	1	0	0
Doc14	Negative	0	1	2	0
Doc15	Negative	2	1	0	0
Doc16	Negative	0	0	1	2

Table II elaborates on each term count concerning its class and count; here, the word is replaced with the term frequency.

From color to frequency, the text label was renamed red as F1, blue as F2, green as F3, and yellow as F4.

Table III describes the total number of term counts for each term in the positive class documents.

TABLE III. TERM FREQUENCY OF POSITIVE CLASS

Class	TC	F1	F2	F3	F4
Positive	1	4	5	5	5
Positive	2	3	2	3	2
Positive	3	0	0	0	0

Table IV describes the total number of terms counted for each term in the negative class documents.

TABLE IV. TERM FREQUENCY OF NEGATIVE CLASS

Class	TC	F1	F2	F3	F4
Negative	1	2	4	4	3
Negative	2	1	1	2	1
Negative	3	0	0	0	0

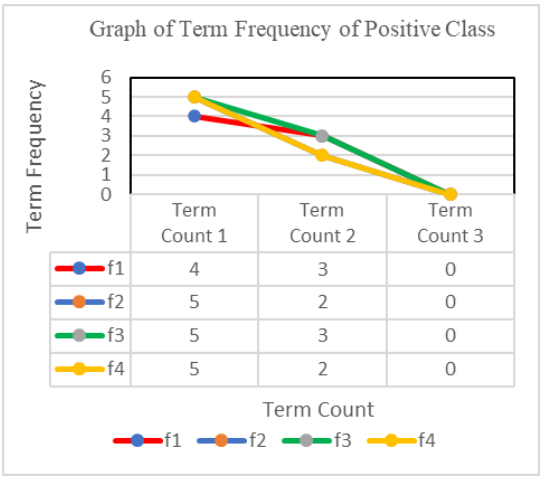

Fig. 3. Graph of term frequency of the positive class.

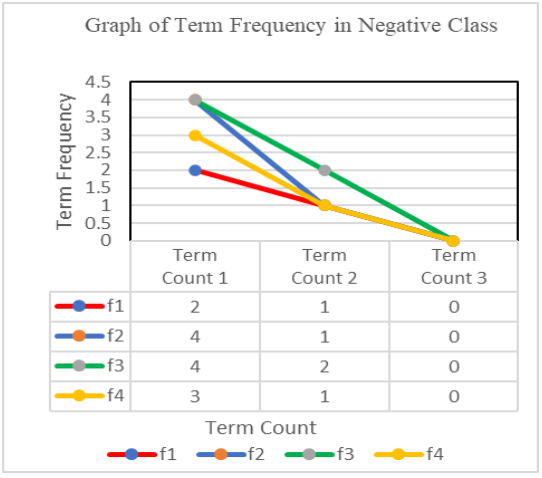

Fig. 4. Graph of term frequency of the negative class.

Fig. 3 to 8 show each term's frequency graph in different classes. Table V elaborates on the total number of term counts for each positive and negative class.

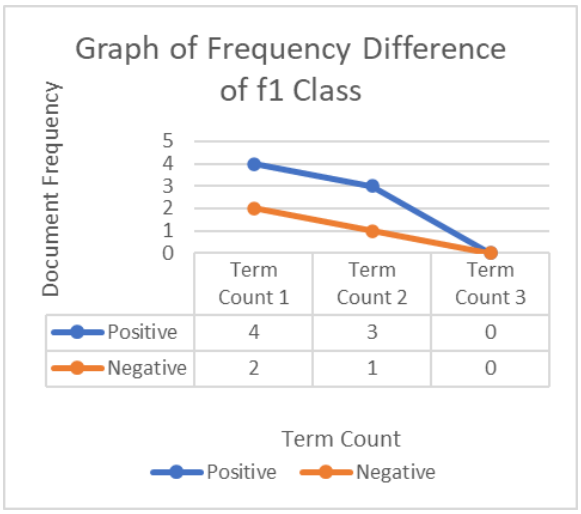

Fig. 5. Graph of frequency difference of the F1 class.

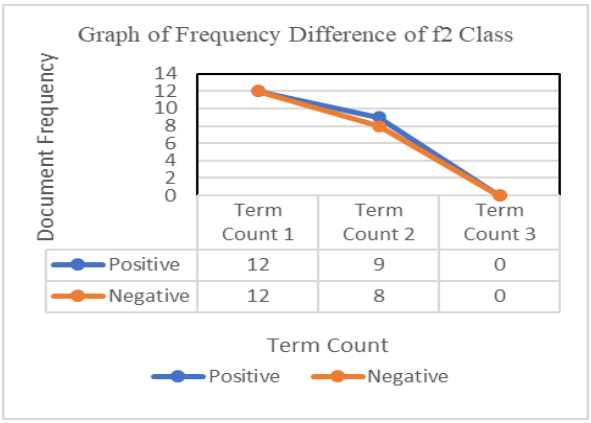

Fig. 6. Graph of frequency difference of the F2 class.

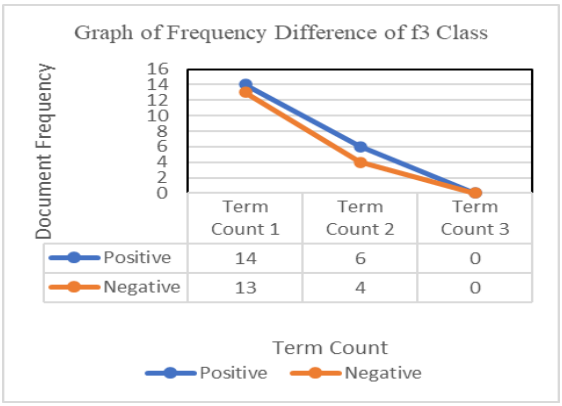

Fig. 7. Graph of frequency difference of the F3 class.

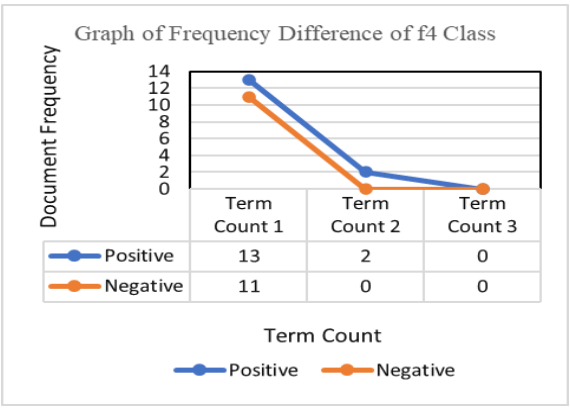

Fig. 8. Graph of frequency difference of the F4 class.

In Table V, P represent the positive and N represent the negative terms. The given positive and negative classes are normalized by dividing each term by the average of the total documents to obtain a normalized term in each positive and negative class, as described in Table VI.

$$u = \sum_{i=1}^{n} \left(\frac{n}{An}\right)$$

TABLE V. FREQUENCY DIFFERENCE OF EACH FREQUENCY IN EACH CLASS

Term count	F1		F2		F4		F4	
	P	N	P	N	P	N	P	N
1	4	2	5	4	5	4	5	3
2	3	1	2	1	3	2	2	1
3	0	0	0	0	0	0	0	0

TABLE VI. TERM FREQUENCY OF POSITIVE AND NEGATIVE CLASS

Term count	F1		F2		F3		F4	
	P	N	P	N	P	N	P	N
1	0.40	0.20	0.416	0.333	0.357	0.285	0.454	0.272
2	0.30	0.10	0.166	0.083	0.214	0.142	0.181	0.090
3	0	0	0	0	0	0	0	0

TABLE VII. COUNT DIFFERENCE OF TERM FREQUENCIES

Term Count (tc)	P (Tprtc)	N (Fprtc)	Difference (D)	Minimum (γ)	BRDC = (D/γ) times (AUCt = Sum +(BRDCtc+i/2)h)
F1					
1	0.4	0.2	0.2	0.2	**4.5**
2	0.3	0.10	0.2	0.1	
F2					
1	0.416	0.333	0.083	0.333	**2.125**
2	0.166	0.083	0.083	0.083	
F3					
1	0.357	0.285	0.071	0.287	**1.125**
2	0.214	0.142	0.071	0.142	
F4					
1	0.454	0.272	0.181	0.272	**2.333**
2	0.181	0.090	0.090	0.090	

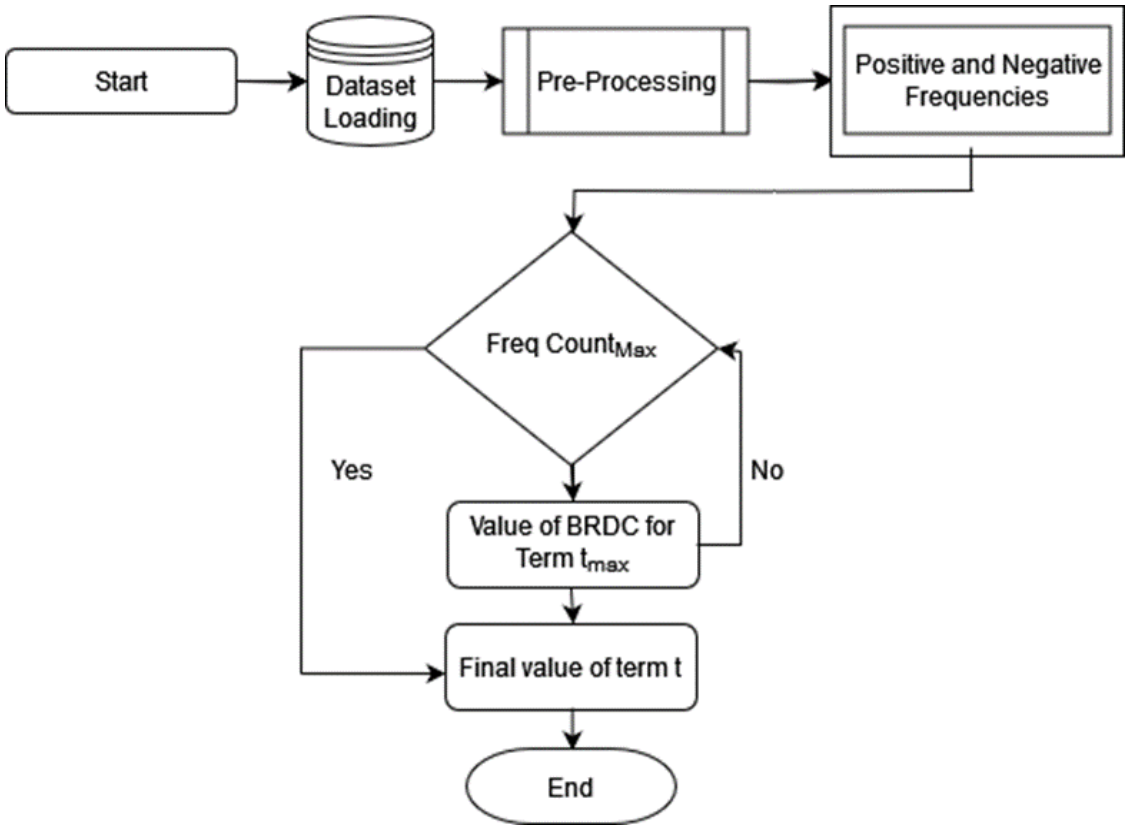

Fig. 9. Proposed BRDC model.

Table VII shows the calculated values of BRDC. Fig. 9 demonstrates the working flow of classification steps with the proposed feature ranking.

TABLE VIII. COMPARISON RESULTS OF RDC, IRDC AND BRDC

Technique	F1	F2	F3	F4
RDC	1.50	0.625	0.375	0.833
IRDC	0.226	0.274	0.095	0.096
BRDC	4.50	2.125	1.125	2.333

Table VIII shows the results of the existing and proposed feature ranking techniques. In contrast to the research conducted by study [28] and [43], this study assigns a rank according to the rarely and frequently occurring significant terms in each class for classification efficiency. It does the trade between both terms, and it does the trade between exploration and exploitation. It also reduces the complexity of the proposed algorithms, such as IRDC and RDC. It reduces the complexity and can work more efficiently for time series-based datasets by,

- Calculate the term counting a normalized technique
- Calculation of the BRDC by reducing the iteration
- Calculating the BRDC using the integral method

$$BRDC = [(TPR_{tc} - FPR_{tc}) / \min(TPR_{tc}, FPR_{tc})] * tc$$

$$AUC_t = Sum + (BRDC_{tc-i}/2)\, h \quad (1)$$

The proposed algorithm pseudo is given below.

Start
Stage 1. Insert: text dataset
Stage 2. Preprocess the dataset
Stage 3. Conversion: tf matrix
Stage 4: Number of docs in +ev class and -ev class
Stage 5: u = $\sum_{i=1..n}^{n}(\frac{n}{Ac})$
Stage 6: $MAX_{tc:}$ Maximum count for a term count t
Stage 7: n represents a term. Ac represents the average of total terms
Stage 8: find the discriminant, calculating the Discriminant value to normalize it in stage 9
for tc =1 to $MAX_{tc\ (n)}$ do tc++
topic = documents containing the term t having term count tc in the positive class
fptc = documents containing the term t having term count tc in the negative class
TPR_{tc} = TPtc(i) /u
FPRtc = FPtc(i) /u
BRDC= [(TPRtc-FPRtc) / min (TPRtc, FPRtc)] *tc
AUCt = Sum+(BRDCtc,i/2)h
end for loop
end

B. Mathematical of Definite Integral base Calculation AUC Methodology

Here, we apply the trapezoidal method to calculate the area under the curve (AUC) of a definite integral using trapezoids [44], which can also manage nonlinear or time-series data compared with RDC and IRDC.

$$\frac{a+b}{2} \times h\ (two\ trapezoidal\ ) \tag{2}$$

If we have a continuous function between a specific interval to calculate the area, it will be defined as,

$$\int_a^b f(x)dx \tag{3}$$

Suppose f(x) is a continuous function with an interval of (a, b). Now divide the intervals (a, b) into n equal sub-intervals with each of width,

$$\text{such that } \Delta x = (b-a)/n, \text{suchthata} = x0 < x1 < x2 < x3 < \cdots .. < xn = b \tag{4}$$

Next, the area approximation of the definite integral using the Trapezoidal Rule

$\int_a^b f(x)dx$, is given as in below

$$\int_a^b f(x)dx \approx Tn = \triangle x/2[f(x0) + 2f(x1) + 2f(x2) + \cdots .2f(xn-1) + f(xn)] \tag{5}$$

where,

$$xi = a + i \triangle x \tag{6}$$

If $n \to \infty$, R. H. S of the expression approaches,

$$\text{thedefiniteintegral} \int_a^b f(x)dx$$

Where n resents the number of trapezoids, and the sub-intervals are demonstrated by [x0, x1] [x0, x1], [x1, x2] [x1, x2], ..., [xn−1, xn] [xn−1, xn] were,

$$x0 = a$$

$$x1 = a + \Delta x x1 = a + \Delta x$$

$$x2 = x1 + \Delta x x2 = x1 + \Delta x \ ....$$

$$xn-1 = xn-2 + \Delta x xn-1 = xn-2 + \Delta x$$

$$xn = xn-1 + \Delta x xn = xn-1 + \Delta x \tag{7}$$

Similar to text classification, the term count can be infinite depending on the nature of the corpus or its document(s). The proposed method counts the term count of infinite terms with more accurate results (s) for time-series data.

Here, is the proof of estimation using the integral method.

The area under the curve, such as that in the top character, was divided into trapezoids to demonstrate the trapezoidal rule. This step is proposed to perform well for time series-based datasets. The height of the first trapezoid is Δx, and its parallel bases have lengths y0 or f(x0), and y1 or f1. Therefore, the area of the first trapezoid in can be expressed as

$$(1/2)\Delta x[f(x0) + f(x1)] \tag{8}$$

The areas of the next trapezoids will be as $(1/2)\Delta x[f(x1) + f(x2)]$, $(1/2)\Delta x[f(x2) + f(x3)]$, and so on.

Therefore,

$$\int_a^b f(x)\,dx \approx (1/2)\Delta x\,(f(x0) + f(x1)) + (1/2)\Delta x\,(f(x1) + f(x2)) + (1/2)\Delta x\,(f(x2) + f(x3)) + \ldots + (1/2)\Delta x\,(f(n-1) + f(xn)) \tag{9}$$

Next, taking out a common factor of (1/2) Δx and combining like terms, we have,

$$\int_a^b f(x)\,dx \approx (\Delta x/2)\,(f(x0) + 2f(x1) + 2f(x2) + 2f(x3) + \ldots + 2f(n-1) + f(xn)) \tag{10}$$

C. Steps to Proceed with the AUC

Four significant steps are involved in calculating the proposed normalized technique. The mentioned steps describe the application of a normalized form of a given curve, y = f(x).

- Step 1: list out the total number of sub-intervals “n”.
- Step 2: List out the interval “a” and “b”.
- Step 3: Calculate the sub-interval using the formula, width, h (or) △x = (b - a)/n.
- Step4: To find the approximation of area (a normalized form of given data) substitute the obtained values in the trapezoidal rule formula,

D. Classification

Widely used classifiers for text classification, such as DT, MNB, LR, and LSTM, are used in this study. These classifiers were selected based on their effective performance in text classification challenges [45-48]. In text classification, LSTM is one of the commonly used deep learning classifiers and a Naïve

Base where Bayes theorem's probabilistic principles underpin the operation of this classifier. It predicts the class of a new sample by evaluating its association with each class, and classifies cases according to how similar they are in that class [49]. One study introduced sentiment analysis as a subfield of information retrieval and computational linguistics, focusing on evaluating the sentiment expressed in text. This study proposes a method for feature selection in sentiment analysis using decision trees, which are evaluated using a Rating System dataset, with preliminary results showing promise [50].

Using the given training data, the DT machine-learning algorithm builds a hierarchical structure and learns basic decision rules to predict the estimated value of a given value. To produce a structure resembling a tree, it recursively divides the feature space according to the values of input features. A decision rule based on a particular feature is specified at each internal node of the tree, and the tree branches out of these. Finally, the leaf nodes of the decision tree deliver the estimated target values based on the patterns discovered during the training [51].

We test the proposed solution using WEKA-3.8.4 (Waikato Environment for Knowledge Analysis), a known machine learning toolkit. All models were tested with a default parameter setting [52]; WEKA was developed using Java, a General Public License (GPL)-based software with different model prediction purposes. In the WEKA toolkit, different iterations are the default numbers required to yield statistically significant results.

E. Experimental Setup

Experiments on the proposed BRDC feature-ranking technique were conducted using an HP workstation machine Z-440 Xeone with 32 GB of RM, and the WEKA tool was used for classification and evaluation purposes. Accuracy, precision, recall, and F-measure were used to evaluate the performance of the proposed approach and compare it with existing approaches. The results demonstrated that the BRDC technique outperformed existing feature-ranking techniques such as RDC and IRDC. Different classes from three benchmark text datasets, Reuters-21578, 20newsgroup, and AG news, were used to evaluate the performance of these feature-ranking approaches. We performed tests with two benchmark datasets that have been utilized in previous experimental studies: [28] and [43], named datasets Reuter21578, 20newsgroup, and another news AG News data. These datasets were extracted and made available for UCI data collection. Fifteen skewed-size classes were obtained from the Reuters-21578 dataset. There is another dataset, 20newsgroup, which has 20 sizable classes and is balanced, and the AG news consists of four classes. All datasets used in this study were labeled in their classes. In addition to word stemming, a stop word list was used to eliminate stop words.

The true positive (TP), false positive (FP), true negative (TN), and false negative (FN) values from the confusion matrix were used to calculate the performance metrics of the algorithms. F1-Score, Accuracy, Precision, and Recall were among the calculated parameters.

$$\text{Accuracy} = \frac{tp + tn}{tp+tn+fn+fp} \tag{12}$$

Precision calculated as follows:

$$Precision = \frac{tp}{tp+ fp} \tag{13}$$

In the above equations, where tp represents the value of the true positive rate, and the false positive rate is represented by fp in terms of accuracy and precision, the value of recall is calculated as follows:

$$Recall = \frac{tp}{tp+ fn} \tag{14}$$

where tp defines the true positive rate and fn represents the false-negative rate in recall.

$$F - Measure = \frac{2\times P\times R}{P+R} \tag{15}$$

Where tp defines the true positive rate and fn represents the false-negative rate in the recall.

IV. Experiments and Results

This section compares the proposed BRDC algorithm with the performance of two existing feature-ranking algorithms, RDC and IRDC. Three text datasets, Reuter-21578, 20newsgroup, and AG News, available at the Kaggle and UCI responses, were used to evaluate the performance of the proposed BRDC algorithm and compare it with RDC and IRDC. They were executed sequentially on a PC running HP workstation z-440 with 32GB RAM for the main system. Furthermore, the number of features chosen and the performance of the classifiers were verified based on the accuracy, precision, recall, and F-measure measuring matrix.

A. Results Using Reuters21578

These datasets, sourced from the UCI library, were used in the experiments. Following the experiments, the Relative Discriminative Criterion (RDC) and Improved Relative Discriminative Criterion (IRDC) were used to compare results. The effectiveness of these feature ranking algorithms was investigated using three distinct datasets: Reuters21578, 20newsgroup, and AG News, and several tests were carried out on the 10, 20, 50, 100, 200, 500, 1000, and 1500 features chosen from the aforementioned datasets. These datasets, sourced from the UCI library, were used in the experiments. The results for Reuter21578 are summarized in Table IX.

Table IX demonstrates the results of the Reuters dataset, which was used to evaluate the performance of the BRDC feature ranking compared with RDC and IRDC using the Reuters-21578 dataset. Classifiers, such as DT, LR, MNB, and LSTM, were employed for this comparison.

Fig. 10 provides a graphical view of the results, showing that BRDC outperforms the other methods in accuracy, precision, recall, and F-measure using the Reuters-21578 dataset. It demonstrates an accuracy of 66.66%, 66.66%, 60.00%, and 73.33%, while it achieves precision of 70.70%, 67.30%, 61.50% and 83.00%, recall of 66.70%, 66.70%, 60.00% and 71.30% and F-measure 65.80%, 66.70%, 59.60% and 70.90% against DT, LR, MNB, and LSTM, respectively. The results indicated that LSTM outperformed DT, LR, and MNB.

TABLE IX. RESULT OF REUTERS-21578 DATASET: A COMPARATIVE ANALYSIS OF BRDC

Technique	Measuring Matrix	DT	LR	MNB	LSTM
BRDC	Accuracy	66.66%	66.66%	60.00%	73.33%
	Precision	70.70%	67.30%	61.50%	83.00%
	Recall	66.70%	66.70%	60.00%	71.30%
	F-Measure	65.80%	66.70%	59.60%	71.90%
IRDC	Accuracy	55.51%	54.30%	54.08%	70.06%
	Precision	54.50%	54.50%	54.00%	64.80%
	Recall	55.50%	54.50%	54.10%	70.10%
	F-Measure	54.55%	54.40%	53.90%	60.00%
RDC	Accuracy	45.50%	44.48%	43.00%	61.60%
	Precision	44.50%	44.40%	43.09%	51.70%
	Recall	45.50%	44.50%	43.00%	61.60%
	F-Measure	44.50%	44.30%	42.80%	61.00%

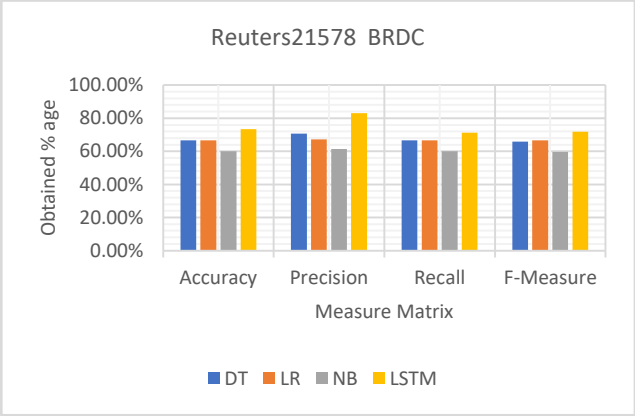

Fig. 10. BRDC with Reuters-21578.

Fig. 11, extracted from Table IX, using the Reuters-21578 dataset, shows the results of the IRDC. It achieves an accuracy of 55.51%, 54.30%, 54.08%, and 70.06%, precision of 54.50%, 54.50%, 54.00%, and 64.80%, recall of 55.50%,54.50%, 54.10%, and 70.10%, and it achieve F-measure of 54.55%, 54.40%, 53.90% and 60.00% against DT, LR, MNB, and LSTM, respectively, however, these results are lower than that of BRDC, here it also shows that IRDC perform better against LSTM.

Fig. 12, which is mined from Table IX, shows the results of the RDC using the Reuters21578 dataset. The results show that RDC achieves an accuracy of 45.50%, 44.48%, 43.00%, and 61.60%; precision of 44.50%, 44.40%, 43.09%, and 51.70%; recall of 45.50%, 43.40%, 43.00%, and 61.60%; and F-measure of 44.50%, 44.30%, 42.80%, and 61.00%, against the DT, LR, MNB, and LSTM models, respectively.

B. *Experiment Using 20newsgroup*

The performance of the proposed BRDC on 10 different classes from the 20newsgroup dataset was analysed based on accuracy, precision, recall, and F-measure metrics. The experiments demonstrated that BRDC produced superior results to the existing IRDC and RDC feature ranking techniques. Table X presents an evaluation of the 20newsgroup datasets using the different classifiers.

Fig. 13 provides a graphical view of the results, showing that BRDC outperformed the other methods in terms of accuracy, precision, recall, and F-measure using the dataset of 20newsgroup. It demonstrates an accuracy of 41.44%, 33.33%, 31.53%, and 50.54%, while it achieved a precision of 32.10%, 30.20%, 28.10%, and 50.70%, recall of 41.40%, 33.30%, 31.50%, and 50.50%, respectively, and F-measures of 33.8%, 31.30%, 29.10%, and 50.60% against DT, LR, MNB, and LSTM, respectively. The results indicated that LSTM outperformed DT, LR, and MNB.

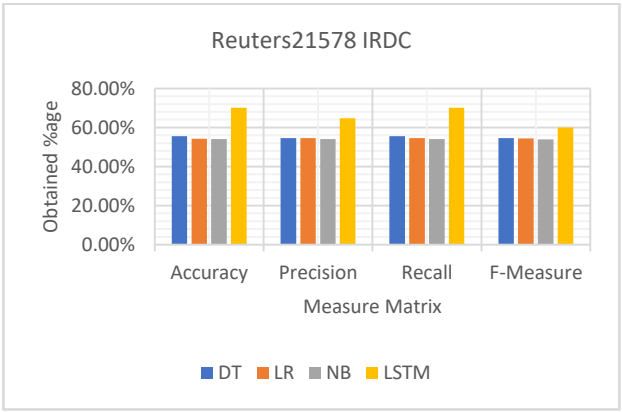

Fig. 11. IRDC with Reuters-21578.

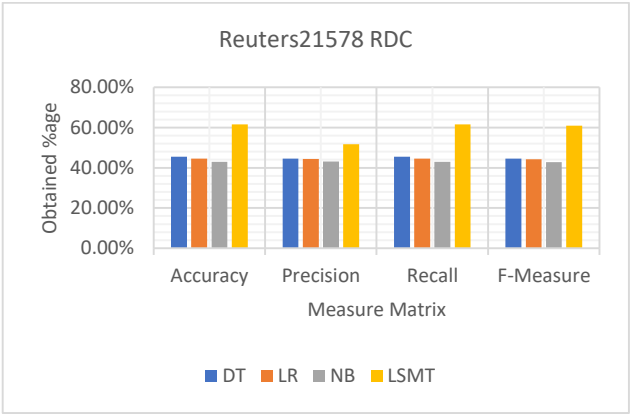

Fig. 12. RDC with Reuters-21578.

TABLE X. RESULT OF 20NEWSGROUP DATASET

Technique	Measuring Matrix	DT	LR	MNB	LSTM
BRDC	Accuracy	45.94%	37.83%	38.73%	50.54%
	Precision	41.50%	37.50%	38.10%	50.70%
	Recall	45.90%	37.80%	38.70%	50.00%
	F-Measure	40.10%	37.60%	38.20%	50.60%
IRDC	Accuracy	44.80%	35.30%	36.73%	48.64%
	Precision	40.50%	35.20%	36.30%	48.80%
	Recall	44.20%	34.60%	36.60%	48.60%
	F-Measure	43.10%	34.30%	36.40%	48.60%
RDC	Accuracy	41.30%	33.83%	31.53%	46.84%
	Precision	32.10%	33.50%	28.10%	47.00%
	Recall	41.40%	32.80%	31.50%	46.80%
	F-Measure	33.10%	33.60%	29.10%	46.80%

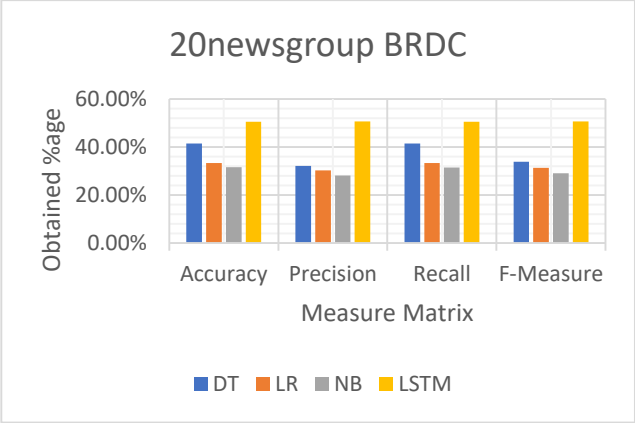

Fig. 13. BRDC with 20Newsgroup.

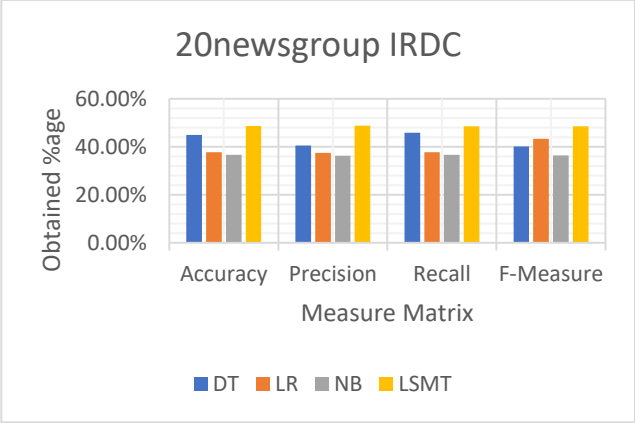

Fig. 14. RDC with 20Newsgroup.

Fig. 14 provides a graphical view of the results, showing that IRDC outperformed the 20newsgroup datasets in terms of accuracy, precision, recall, and F-measure. It demonstrated an accuracy of 44.94%, 37.83%, 36.73%, and 48.64%, respectively, while it achieved a precision of 40.50%, 37.50%, 36.30%, and 48.80%, recall of 45.90%, 37.80%, 36.70% and 48.60%, respectively, and F-measures of 40.10%, 43.30%, 36.40%, and 48.60% against DT, LR, MNB, and LSTM, respectively. Fig. 15, extracted from Table VIII, shows the accuracy results obtained using RDC. It achieves an accuracy of 45.94%, 37.83%, 38.73% and 46.84%, precision of 41.50%, 37.50%, 38.10% and 47.00%, recall of 45.90%, 37.80%, 38.70% and 46.80%, and F-measure of 40.10%, 37.60%, 38.20% and 46.80% against DT, LR, MNB, and LSTM, respectively, however these results are lower than that of BRDC, here it also shows that IRDC performs better against LSTM.

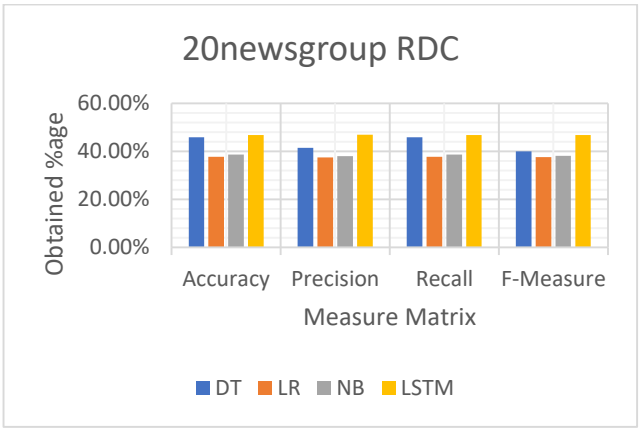

Fig. 15. RDC with 20Newsgroup.

C. Experiment Using AG News

Following the experiments, the Relative Discriminative Criterion (RDC) and improved Relative Discriminative Criterion (IRDC) were used to compare results. Table XI shows the proposed BRDC experiments and compares IRDC and RDC techniques using four classifiers: decision tree, logistics regression, multinomial naïve Bayes, and long short-term memory. The results show the deep learning model outperforms the other machine learning models.

TABLE XI. RESULT OF AG NEWS DATASET

Technique	Measuring Matrix	DT	LR	MNB	LSTM
BRDC	Accuracy	45.09%	44.48%	45.69%	56.09%
	Precision	44.90%	44.50%	45.60%	56.10%
	Recall	45.10%	44.50%	45.70%	56.10%
	F-Measure	47.70%	44.40%	45.50%	56.00%
IRDC	Accuracy	42.10%	43.70%	42.70%	51.10%
	Precision	41.95%	43.40%	42.61%	51.11%
	Recall	42.20%	43.00%	42.71%	51.11%
	F-Measure	42.69%	43.20%	42.49%	51.01%
RDC	Accuracy	40.20%	42.30%	40.60%	50.00%
	Precision	40.90%	42.40%	40.60%	50.08%
	Recall	40.10%	42.40%	40.70%	50.10%
	F-Measure	40.70%	42.20%	40.50%	50.00%

Fig. 16 provides a graphical view of the results, showing that BRDC outperforms the other methods in terms of accuracy, precision, recall, and F-measure. It demonstrated an accuracy of 45.09%, 44.48%, 45.69%, and 56.09%, while it achieved a precision of 44.90%, 43.50%, 45.60%, and 56.10%, recall of 45.10%, 44.50%, 45.70%, and 56.10%, and F-measure of 47.70%, 44.40%, 45.50%, and 56.00% against DT, LR, MNB, and LSTM, respectively using AG news dataset. The results indicated that LSTM outperformed DT, LR, and MNB.

Fig. 17 provides a graphical view of the results, showing that IRDC outperformed the other methods in terms of accuracy, precision, recall, and F-measure. It demonstrates accuracy of 42.10%, 43.70%, 42.70%, and 51.10%, while it achieves a precision of 41.95%, 43.40%, 42.61% and 51.11%, recall of 42.20%, 43.00%, 42.71%, and 51.11%, it achieves F-measure of 42.69%, 43.00%, 42.49% and 51.01% against DT, LR, MNB, and LSTM, respectively using AG news. The results indicated that LSTM outperformed DT, LR, and MNB.

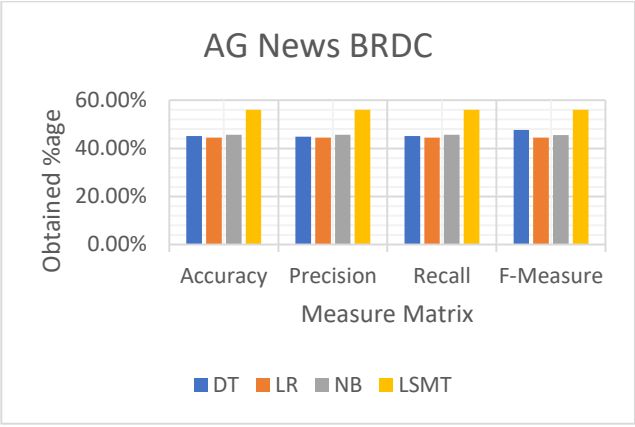

Fig. 16. BRDC with AG News.

Fig. 18 shows a graphical view of the results, showing that RDC outperformed the other methods in terms of accuracy, precision, recall, and F-measure. It demonstrates the accuracy of 40.20%, 42.30%, 40.60%, and 50.00%, while it achieves a precision of 40.90%, 42.40%, 40.60%, and 50.08%, recall of 40.10%, 42.40%, 40.70% and 50.10% and F-measure of 40.70%, 42.20%, 40.50% and 50.00% against DT, LR, MNB, and LSTM, respectively. The results indicated that LSTM outperformed DT, LR, and MNB.

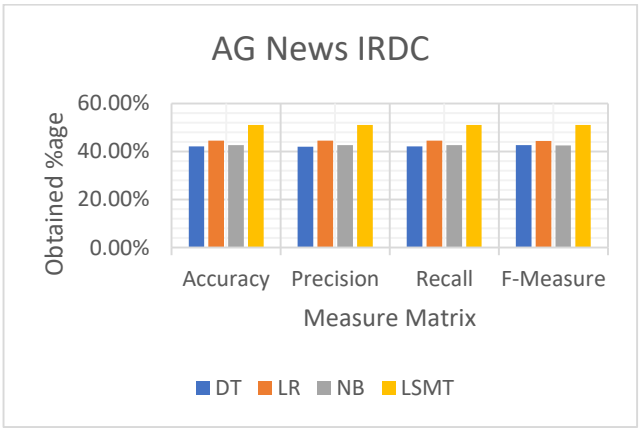

Fig. 17. RDC with AG News.

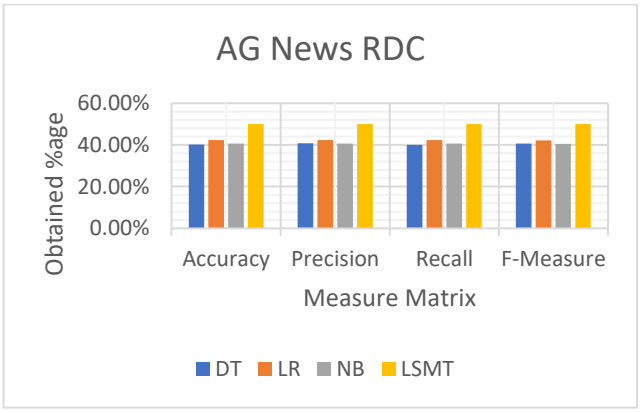

Fig. 18. IRDC with AG News.

V. CONCLUSION

Dealing with high-dimensional text data poses a challenging problem for machine-learning algorithms. This research focuses

on feature ranking and reducing the number of unnecessary and duplicate features to enhance the classifier performance, especially for text. It highlights the limitations of the existing feature ranking techniques, such as RDC and IRDC. To address shortcomings in existing studies, this study proposed the BRDC approach, which was tested on balanced and unbalanced text datasets. The key contribution of the proposed BRDC technique is to adjust the true-positive and false-positive rates for term counts in the positive and negative classes in a balanced way, ranking for both frequently and rarely occurring terms and term counts in both classes, using a balanced normalized approach. The BRDC considers common and infrequent terms and normalizes them to improve classification accuracy. Compared to RDC and IRDC, BRDC selects optimal features and enhances classification performance. The proposed approach also reduces the number of iterations to calculate the AUC and uses an integral-based approach. Additionally, the proposed approach is compared with different machine learning and deep learning models, which shows that deep learning models outperform machine learning models.

We will discuss how the proposed technique affects balanced and unbalanced image datasets in future work. Use other integral-based methods to calculate AUC. In addition, we aim to evaluate the proposed integral-based approach for different image datasets and other integral-based methods such as Simpson's based approach. We are also planning to review the temporal demands of the proposed model using different textual and image datasets.

REFERENCES

[1] Raghavan, P., Text Centric Structure Extraction and Exploitation (abstract only), in Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004. 2004, Association for Computing Machinery: Paris, France. p. 0.

[2] Rehman, A., et al., Relative discrimination criterion – A novel feature ranking method for text data. Expert Systems with Applications, 2015. 42.

[3] Malik, S. and S. Jain, Deep Convolutional Neural Network for Knowledge-Infused Text Classification. New Generation Computing, 2024.

[4] Al-Fuqaha'a, S., N. Al-Madi, and B. Hammo, A robust classification approach to enhance clinic identification from Arabic health text. Neural Computing and Applications, 2024. 36(13): p. 7161-7185.

[5] Labani, M., et al., A novel multivariate filter method for feature selection in text classification problems. 2018. 70: p. 25-37.

[6] Ahmad, N. and A. Nassif, Dimensionality Reduction: Challenges and Solutions. ITM Web of Conferences, 2022. 43: p. 01017.

[7] Torkkola, K., Discriminative features for document classification. Vol. 1. 2002. 472-475 vol.1.

[8] Haq, A.U., et al., Combining Multiple Feature-Ranking Techniques and Clustering of Variables for Feature Selection. IEEE Access, 2019. 7: p. 151482-151492.

[9] Kowsari, K., et al., Text Classification Algorithms: A Survey. 2019. 10(4): p. 150.

[10] Zubair, I.M. and B. Kim, A Group Feature Ranking and Selection Method Based on Dimension Reduction Technique in High-Dimensional Data. IEEE Access, 2022. 10: p. 125136-125147.

[11] Xiao, H., Application of Digital Information Technology in Book Classification and Quick Search in University Libraries. Comput Intell Neurosci, 2022. 2022: p. 4543467.

[12] NAQVI, S., A Hybrid filter-wrapper approach for FeatureSelection. 2011.

[13] Ladha, L., T.J.I.j.o.c.s. Deepa, and engineering, Feature selection methods and algorithms. 2011. 3(5): p. 1787-1797.

[14] Forman, G., An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res., 2003. 3(Mar): p. 1289-1305.

[15] Azam, M., et al., Feature Extraction based Text Classification using K-Nearest Neighbor Algorithm. 2018.

[16] Dzisevič, R. and D. Šešok. Text Classification using Different Feature Extraction Approaches. in 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream). 2019.

[17] Biricik, G., B. Diri, and A. SÖNmez, Abstract feature extraction for text classification. Turkish Journal of Electrical Engineering and Computer Sciences, 2012. 20: p. 1137-1159.

[18] Harrag, F., E. El-Qawasmah, and A.M.S. Al-Salman. Stemming as a feature reduction technique for Arabic Text Categorization. in 2011 10th International Symposium on Programming and Systems. 2011.

[19] Bharti, K.K. and P.K. Singh, Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. Expert Systems with Applications, 2015. 42(6): p. 3105-3114.

[20] Dash, M. and H.J.I.d.a. Liu, Feature selection for classification. 1997. 1(1-4): p. 131-156.

[21] Bolón-Canedo, V., et al., A review of feature selection methods on synthetic data. 2013. 34: p. 483-519.

[22] Jović, A., K. Brkić, and N. Bogunović. A review of feature selection methods with applications. in 2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO). 2015. Ieee.

[23] Jin, X., et al. Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles. in Data Mining for Biomedical Applications: PAKDD 2006 Workshop, BioDM 2006, Singapore, April 9, 2006. Proceedings. 2006. Springer.

[24] Hunt, E.B., J. Marin, and P.J. Stone, Experiments in induction. 1966.

[25] Chandrashekar, G. and F. Sahin, A survey on feature selection methods. Computers & Electrical Engineering, 2014. 40(1): p. 16-28.

[26] Dash, M., et al. Feature selection for clustering - a filter solution. in 2002 IEEE International Conference on Data Mining, 2002. Proceedings. 2002.

[27] Rehman, A., et al., Relative discrimination criterion–A novel feature ranking method for text data. 2015. 42(7): p. 3670-3681.

[28] Sharif, W., et al., Improved relative discriminative criterion feature ranking technique for text classification. International Journal of Artificial Intelligence, 2017. 15: p. 61-78.

[29] Jin, L. and L. Zhang. De-redundancy Relative Discrimination Criterion-based Feature Selection for Text Data. in 2022 International Joint Conference on Neural Networks (IJCNN). 2022.

[30] Hemmati, M., et al. A New Hybrid Method for Text Feature Selection Through Combination of Relative Discrimination Criterion and Ant Colony Optimization. 2022. Singapore: Springer Nature Singapore.

[31] Li, B., et al. Weighted Document Frequency for feature selection in text classification. in 2015 International Conference on Asian Language Processing (IALP). 2015.

[32] Baccianella, S., A. Esuli, and F.J.E.S.w.A. Sebastiani, Using micro-documents for feature selection: The case of ordinal text classification. 2013. 40(11): p. 4687-4696.

[33] Silva, W.A. and S.M. Villela, Improving the one-against-all binary approach for multiclass classification using balancing techniques. Applied Intelligence, 2021. 51(1): p. 396-415.

[34] Fesseha, A., et al. Text Classification of News Articles Using Machine Learning on Low-resourced Language: Tigrigna. in 2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD). 2020.

[35] Parlak, B. and A.K. Uysal, A novel filter feature selection method for text classification: Extensive Feature Selector. 2023. 49(1): p. 59-78.

[36] Ige, O.P. and G. Keng Hoon, Ensemble feature selection using weighted concatenated voting for text classification. Journal of Nigerian Society of Physical Sciences, 2024. 6(1): p. 1-8.

[37] Nachaoui, M., I. Lakouam, and I. Hafidi, Hybrid particle swarm optimization algorithm for text feature selection problems. Neural Computing and Applications, 2024. 36(13): p. 7471-7489.

[38] Azam, N. and J. Yao, Comparison of term frequency and document frequency based feature selection metrics in text categorization. Expert Systems with Applications, 2012. 39(5): p. 4760-4768.

[39] Jin, C., et al., Chi-square Statistics Feature Selection Based on Term Frequency and Distribution for Text Categorization. IETE Journal of Research, 2015. 61(4): p. 351-362.

[40] Zhen, Z., et al. Categorical Document Frequency Based Feature Selection for Text Categorization. in 2011 International Conference of Information Technology, Computer Engineering and Management Sciences. 2011.

[41] Poolsawad, N., C. Kambhampati, and J. Cleland. Balancing class for performance of classification with a clinical dataset. in proceedings of the World Congress on Engineering. 2014.

[42] Yolchuyeva, S., G. Németh, and B. Gyires-Tóth, Text normalization with convolutional neural networks. International Journal of Speech Technology, 2018. 21(3): p. 589-600.

[43] Rehman, A., et al., Relative discrimination criterion – A novel feature ranking method for text data. Expert Systems with Applications, 2015. 42(7): p. 3670-3681.

[44] Wright, M., Entering the era of computationally driven drug development. Drug Metabolism Reviews, 2020. 52.

[45] Al Essa, A., Efficient Text Classification with Linear Regression Using a Combination of Predictors for Flu Outbreak Detection. 2018, University of Bridgeport.

[46] Charbuty, B., A.J.J.o.A.S. Abdulazeez, and T. Trends, Classification based on decision tree algorithm for machine learning. 2021. 2(01): p. 20-28.

[47] Xu, S.J.J.o.I.S., Bayesian Naïve Bayes classifiers to text classification. 2018. 44(1): p. 48-59.

[48] Nowak, J., A. Taspinar, and R. Scherer. LSTM recurrent neural networks for short text and sentiment classification. in Artificial Intelligence and Soft Computing: 16th International Conference, ICAISC 2017, Zakopane, Poland, June 11-15, 2017, Proceedings, Part II 16. 2017. Springer.

[49] Prasad, J.V.D., et al., Relevant-Based Feature Ranking (RBFR) Method for Text Classification Based on Machine Learning Algorithm. Journal of Nanomaterials, 2022. 2022: p. 1-12.

[50] Suresh, A. and C.J.I. Bharathi, Sentiment classification using decision tree based feature selection. 2016. 9(36): p. 419-425.

[51] Mahdieh, L., et al., A novel multivariate filter method for feature selection in text classification problems. Engineering Applications of Artificial Intelligence, 2018. 70: p. 25-37.

[52] Hall, M., et al., The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 2009. 11(1): p. 10-18.