Section A -Research paper

DOCUMENT CLASSIFICATION SYSTEM USING IMPROVISED RANDOM FOREST CLASSIFIER

Kavyashree Nagarajaiah
Assistant Professor
Department of MCA
SSIT, Tumkur
Kavyashree1283@gmail.com

Madhu Hanakere Krishnappa
Associate Professor
Department of MCA
BIT,Bangalore
madhuhkbit@gmail.com

Asha Kempaiah Rukmini
Assistant Professor
Department of MCA
SSIT, Tumkur
ashakr@ssit.edu.in

Abstract

Text classification is the process of categorizing text into pre-established groupings based on its content. The amount of information available on the Internet has grown significantly over the past few years, making the classification of texts one of the most crucial yet difficult tasks. Text classification is frequently used in a wide range of applications and for a variety of purposes. Blogs, Twitter, and other social media platforms are contributing significantly to the exponential growth of textual data on the internet. The type of text that people upload to the internet is not specified by them. The majority of academics in this field are searching for automated solutions to categorize data or give unclassified documents a class designation. One area where texts are classified is text categorization and the researchers offered a number of options for text classification. The methods for classifying the text often involve gathering training data, preparing the text, extracting features, reducing features, representing the document, and then employing classification algorithms to create a model for predicting the class of a new textual document. In this paper, a Document Classification System using an Improvised Random Forest (DCS-IRF) classifier is proposed which attains better performance when compared with other classifiers and the implementation is performed using python toolkit. The DCS-IRF performs using the data obtained from Reuters-21578 dataset, which is a collection of documents with news articles. Moreover, experimental results obtained by using IRF classifier offers excellent results in accuracy and provides efficient classification of the text documents.

Keywords: Document Classification system (DCS), Improvised Random Forest (IRF), Internet, Python toolkit, Reuters-21578, Text classification.

1. Introduction

Text classification is a method to classify text documents based on the traits and characteristics that consist of text data. A common definition of this task as a supervised learning problem is the identification of new document categories based on the likelihoods given by a

predetermined training corpus of previously labelled (identified) documents [1]. Numerous fields, such as the classification of news and the detection of spam require text classification. The idea may appear straightforward, and if there are only a few documents, it is feasible to personally study each one. An automatic approach must be implemented to classify the text documents for the relevant applications. So, the Machine Learning (ML) approach must be applied to classify text documents from the given collection of data [2,3]. The techniques related to ML methods are applied to solve the problems which occur during Natural Language Processing (NLP). The NLP helps in computer programs which translate the text from one language to different language and summarize the large form of text quickly. The ML approach require labeled data to effectively classify the data from the text documents [4,5]. The document classification method is utilized in various fields such financial, agricultural researches, and clinical applications etc [6]. The Document Classification System (DCS) using NLP and ML is very much efficient in classifying the documents in applications related to various fields. The various ML algorithms and NLP techniques are implemented to calculate the overall performance of DCS [7].

The combination of ML and NLP are introduced to classify the documents for various fields of applications. The classification performance of the DCS can be enhanced using the features of Term Frequency-Inverse Document Frequency (TF-IDF) [8]. The classification of documents is performed using the tool kit based on python, which efficiently predicts and classifies the document based on various domains [9]. The documents are collected from Reuters-21578 dataset, which is a collection of documents with news articles, moreover the Reuters-21578 dataset is considered as a benchmark dataset for classifying the documents [10]. The classification of documents is performed for both text and image-based documents, in this paper the classification is performed for text based documents. The text document classification is performed using the improvised random forest classifier which helps to classify the documents accurately while compared with other classifiers such as Naïve Bayes classifier, decision tree classifier and KNN classifier. Moreover, the random forest classifier is utilized for regression, classification and so on. The RFC builds a multitude of decision trees to train the model and classifies the documents accordingly [11]. In this paper, the Document Classification System (DCS) using an Improvised Random Forest (IRF) classifier is proposed to classify the documents efficiently.

The following contributions are made in this paper,

(i) The efficient document classification system is developed which efficiently classifies the document using the proposed IRF classifier.
(ii) The Improved RF classifier is introduced from the existing random forest classifier by improving its capability to reduce the error rate due to correlation between two trees in the forest.
(iii) The proposed IRF classifier model is compared with various existing ML classifier models for categorization of text document approaches.

The remaining structure of the paper of the is structured as follows, the Section.2 is discussed about the related works. The proposed method is presented in Section.3. The Section.4 represents the results obtained from comparison of IRF with various ML classifiers and finally Section.5 represents the summary of the paper.

2. Related Works

Md. Anwar Hussen Wadud *et.al* [12] has introduced an LSTM-BOOST model which uses Adaboost algorithm to perform analysis on principal component combined with the networks of LSTM. The dataset was categorized into three classes in LSTM-boost model and in each part of dataset the networks of PCA and LSTM were applied to get significant variance and to lower the error of the model. The LSTM-Boost model was implemented to classify the text documents which contains offensive text in the social media. However, the LSTM-Boost model classifies only the offensive text and doesn't classifies the text which is contained in the image document.

Anton Thielmann et.al [13] has introduced a one class Support Vector Machine (SVM) and Latent Dirichlet Allocation (LDA) modelling as a multi-step rule for classification. SVM performs unsupervised document classification with the integeration of training data and helps to achieve the correct form of training data. The combination of web scraping, SVM and LDA allows to incorporate the external domain training data and this combination classifies the text documents and avoids the cost and time during manual labeling. However, the difficulty occurs in identifying the text labels that are accurate enough for classification.

Sara Mora et.al [14] have developed a pipeline based on Natural Language Processing (NLP) for extracting information automatically from the report of microbiological culture. At first, the pipeline filters the text to remove the meaningless sentences, then the names of microorganisms are labeled and linked to the set of metadata that contains vocabularies related to international data. Moreover, the pipe line of NLP mines the complete picture from image document. However, due to the shortness of the documents related to clinical records, the records must be in abbreviated form, otherwise the complete category is not included.

Sanda Martinˇci´c-Ipši´c *et.al* [15] has developed a frame work for multi criteria feature representation of text based on classifying the documents. The document representation model combined with the proposed frame work based on representataion of languages of the text documents. The bag of words method is used while classifying the documents and the performance of the document classifier is improved using the random forest method on feature generation and their derivatives. However, during classification of documents the label classes were highly imbalanced that imbalanced state is mainly responsible for the unsteady accuracy value.

Xiaoyu Luo et.al [16] have implemented a SVM model for classifying the documents which contains English texts. The SVM model mainly focuses on providing text classification, feature selection, and evaluation of performance using Weka, a python tool kit. Initially, the pre-processing is performed to select the best features from the English language and then features

were extracted from the documents based on English language in supervised machine learning approach. The text classification using SVM model has the capability to classify any type of documents within or outside the class. However, the SVM model is well suited for classifying the text documents but while coming to the classification of image documents, the SVM model lacks in its overall performance.

Asmaa M. Aubaid et.al [17] have introduced an approach for classification of text documents using embedded techniques. Here, the classification of text documents was done for ten categories for two datasets such as Reuters-21578 and 20 Newsgroups datasets. The approach used python as a tool for programming. The rule based approach provides active results for real time datasets and newspaper datasets. However, the information retrieval system used in this rule based approach provides miscellaneous nature of texts for high variable contents.

3. DCS-IRF classifier for classifying the documents

The overall process of classifying the documents based on text using the proposed DCS-IRF classifier is represented using the block diagram (*Figure.1*) given below,

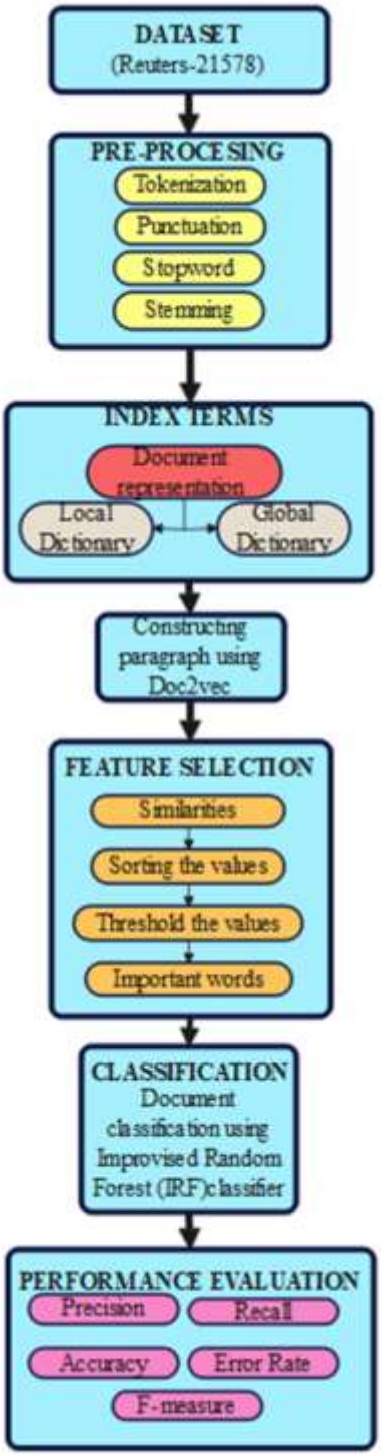


Figure 1 Classification of text document using DCS-IRF classifier

3.1 Dataset

A dataset is generally a group of correlated features of the data-related domain which is utilized in an individual or combined manner. In the dataset, the data is directly connected to a particular information needed. Here, an NLP tool named Document to vector (Doc2vec) is applied in DCS-IRF method using Reuters-21578 dataset. The Reuters-21578 dataset is emerged from Reuters Newswire in the year 1987 which is publicly available dataset used for classification of text documents. The Reuters-21578 dataset is a collection of 22 distributed files, where the 21 files at the initial stage contains 1000 documents and the final file consist of 578 documents. Those 22 distributed files are in SGML format and initiates with, <! DOCTYPE lewis SYSTEM "lewis.DTD"> and the categorization of documents based on dataset is classified into training and testing sets. Every document is classified into five tags such as Topics, Places, Orgs, People, Exchanges [4]. Each category tag consists of numerous topics, but this paper is

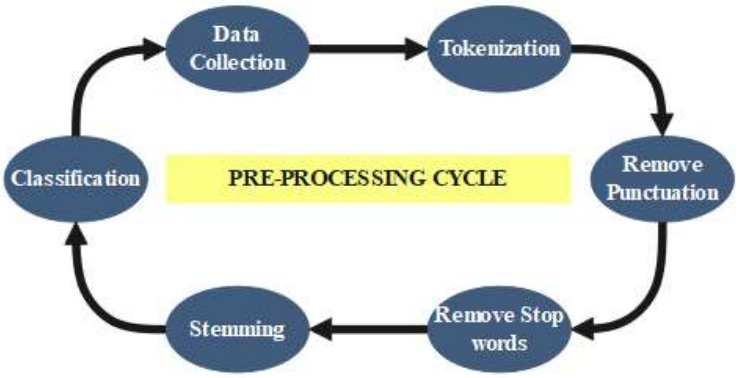


Figure 2 Pre-processing cycle

created using Topic tag category.

3.2 Pre-Processing

The Pre-Processing is a preliminary processing of the data where the primary step for text initialization takes place in a particular processing time. The following process are performed during pre-processing method to improve the quality of raw data. The Pre-Processing of raw data involves various steps such as tokenization, punctuation, removal of stop words and stemming.

3.2.1 Tokenization

Tokenization is the method for dividing text at the input to tokens by keeping track of the order in which tokens appear while concurrently removing certain characters like punctuations. In other words, tokenization is reduction of documents to tokens, which are single words or phrases. When all punctuation is removed from a text and tokenization is used, the entire text gets lowercased.

3.2.2 Punctuation

Punctuation is a set of signs that let words flow naturally and accurately convey their meaning. These markers specify where to pause or give our sentences a symbolic feeling. By separating ideas, punctuation makes sentences pure. Additionally, punctuation is used to emphasize quotes, titles, and other key linguistic elements. The inclusion of punctuation is necessary since it is crucial to any writing. Some examples include ",", "!", “?”, “*”.

3.2.3 Removal of stop words

Eliminating stop words is an important step in text classification. A list of often used words that serve a significant purpose in a text but have no meaning is referred to as a stop word. To cut down on noise terms, elimination of stop words from the text, leaving the keyword undamaged**.** Stop words, including "the," "and," "from," "are," and "to," are frequent words that appear in most of the documents. These stop words will not define the document in the categorization system which are necessary to employ this procedure.

3.2.4 Stemming

When gaining knowledge, stemming reduces the form of the word from the root using certain target language-related rules. This is important because derived words often contain affixes, which include prefixes, infixes, suffixes, and confix. Stemming is the method of getting terms down to the most basic forms. Words like "cultivating," "cultivator," and "cultivated," are shortened to "cultivate," and "decaying" and "decayed" are shortened to "decay." As several word forms are stemmed into an individual word, it helps decrease computation time and space. In actuality, this is the technique' primary benefit in the process of document classification.

3.3 Indexing Terms

The method of indexing is advantageous for categorization procedures and an essential stage in the construction of the lexicon. This dictionary was referred to as a local vocabulary and was used as the primary lexicon for the selection of feature in the classification of text. A separate set of traits from individual class is chosen for this dictionary. The local dictionary policy has been used in a number of studies. The most significant terms from each category were chosen for the local dictionary, which helped to improve the process of classification for each class by choosing contrasting sets of attributes from each independently of the other categories**.**

3.4 Creation of local dictionary

A various set of characteristics from each class is chosen by main dictionary to accomplish selection of feature in the classification of text. This kind of dictionary has been employed in a number of investigations. The local dictionary tries to speed up the categorization process for each class by choosing the most crucial traits in the class. It does this by selecting contrasting sets of features from each, independent of the other class.

3.5 Doc2vec method

The creation of doc2vec model takes place by utilizing the documents of training dataset. Doc2vec method is significant to define the similarity among the words in the language of the text document of the local dictionary as well as training documents to obtain significant selection of feature.

3.5.1 Creation of Doc2vec

At initial stage, Le and Mikolov created word2vec algorithm and later they developed the algorithm named doc2vec which is based on adjusted techniques obtained from word2doc specified for categorization of text. The Doc2vec algorithm is a paragraph-level text

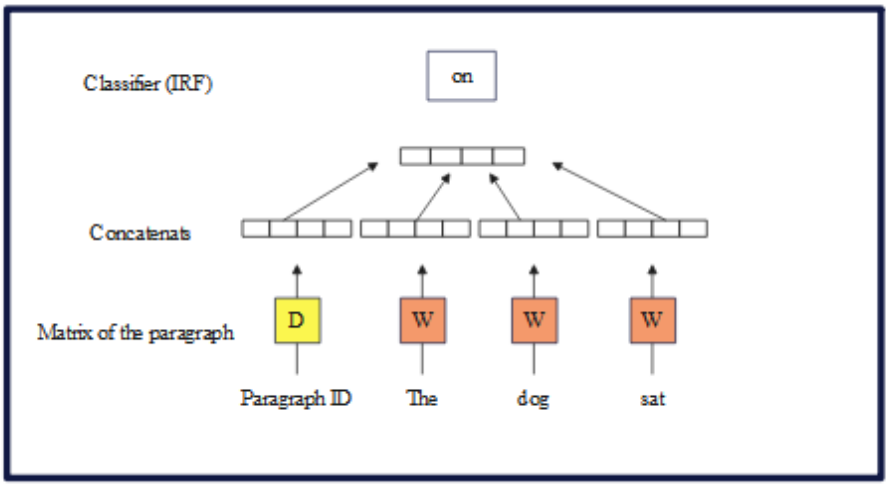


Figure .3 Distributed memory model for doc2vec

categorization algorithm. Moreover, doc2vec is a public tool that is used to categorize text. Doc2vec is a standard method to study the vectors in the word and it is further categorized as two types such as distributed memory model and distributed bag of words model. In our work, distributed memeory model technique is used to learn the vectors of words. In the framework of doc2vec, each class is assigned a distinct vector, that is described by a column in matrix D for each term W.

To predict the next word in a context, the concatenation takes place between the word vector and doc2vec, and the concatenation method is utilized in consolidation of vectors. The contexts are examined from a sliding window over a portion and have a set length. A vector's paragraph is shared across all contexts created from it, but not across the paragraphs. By utilizing the average of the linked vector combined with a context of three words, this model allows for the prediction of the fourth word. The doc2vec is assigned to be absent in the data from context at present and it functions as a memory for the subject in the paragraph. After completion of training, doc2vec is used as vocabularies for paragraph. Finally, there are two stages present in the algorithm: Training stage: At this stage the word vectors W, soft max weights U are acquired from the

observed paragraphs. Inference stage: At this stage doc2vec D is obtained for new paragraphs and gradient descending is performed W, U and keeping b as fixed. D is used to perform prediction on the text labels using the IRF classifier.

3.6 Feature selection

Feature selection is a significant process involved in classification of text documents. The

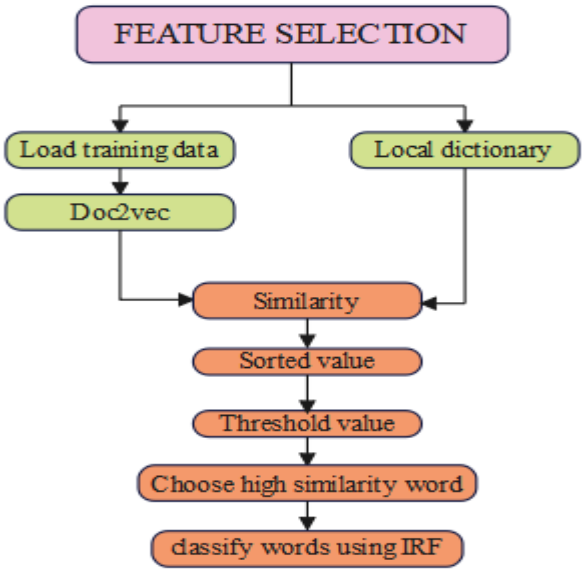


Figure.4 Steps involved in feature selection

text feature selection targets to represent the documents using the suitable features. The feature

selection process can minimize size of the datasets and improve the performance of machine learning algorithms. The feature selection of the text document involves finding the similarities in the text document, sorting the vocabularies in the document, finding the threshold value and classifying the significant text from the document. The steps involved in feature selection is represented in *Figure.4*

3.6.1 Calculating similarities in the document

By training set of data to identify terms which are comparable to have a similar meaning to other words, the vocabulary was recovered from the papers. When trying to avoid using the same term repeatedly, focusing on vocabulary with similarity values close to 1 and eliminating words with similarity values close to 0 (zero) by using a value of threshold can be helpful. Constructing a doc2vec model was used to create texts and calculate the vocabularies which looks similar in local dictionary during the similarity method.

3.6.2 Sorting the vocabularies and finding threshold value of text

The values of vocabulary similarity were arranged in accordance with threshold values, which are thresholds beyond which a program's behavior changes. In particular, the threshold value is defined as measure of how similar phrases were across papers, and this measure was used to identify the key words in those documents. The threshold value is calculated using the median from the similarity values.

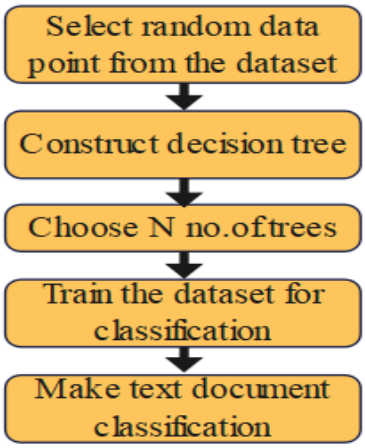
3.6.3 Document classification using IRF

The classification of text documents is performed using the IRF algorithm. Here many decision trees are constructed since they work together and acts as a pillars in the IRF algorithm. Generally, RF is defined as collection of decision trees whose nodes are defined at the step of pre-processing. The greatest feature is chosen from randomly chosen feature subset after many trees have been constructed. Another notion that is created utilizing the decision tree method is to generate a decision tree [18]. These trees make up the random forest, which is utilized to categorize new objects from the vector at the input. Each constructed decision tree is employed during classification. Assuming votes from trees to class, the random forest will select classification that receives the most votes from maximum trees present in the forest. If the normal RF classifier is utilized in document classification, it has drawback such as increased error rate due to correlation between two trees in the forest [19]. To overcome this drawback, the IRF is proposed in this paper which is considered as a best classifier to categorize the documents. In the proposed method, the correlation among two trees are reduced and improve the efficiency during classification of documents. Some features of the IRF classifier is mentioned below.

(i) IRF have the capability to handle large input text document without removal of text in the document
(ii) IRF provides a suggestion to classify the important text and efficiently performs with large databases like Reuters-21578.
(iii) The trees or forest created during the classification process can be utilized for future use.

The representation of work flow of IRF is provided below in *Figure.5*

Figure 5 Work flow of IRF



The improvised random forest classifier can be represented using the mathematical formula,

$$n_{ij} = w_l C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)}$$

Where,

ni sub(j) represents significance of node j

w sub(j) represents the number of weighted samples reaches the node j

C sub(j) represents the value of impurity at node j

left(j) represents left split from child node on node j

right(j) represents right split from child node on node j

4. Results and comparison

The overall performance of the proposed IRF classifier is calculated in this section and the comparative analysis is performed with the existing KNN classifier and Logistic Regression classifier. The comparison between the mentioned machine learning algorithm is performed based on parameters such as precision, F1-score, accuracy, support value.

4.1 Evaluation measures

(i) Accuracy: The accuracy is the ratio of observation predicted suitably to the total observations. In simple words, accuracy is the percentage of correctly classified documents. The mathematical way of representation is denoted as

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{True\ Positives\ +\ True\ Negatives}}{(\boldsymbol{True\ Positives\ +\ True\ Negatives + False\ Positives\ +\ False\ Negatives})}$$

(ii) Precision: Generally precision value is calculated by taking the ratio of observations which are predicted positively true to total positive predicted observations. Here, precision during document classification is designed by percent of relevant documents correctly retrieved with respect to text present in the documents. In mathematical way it is represented as

$$\boldsymbol{Precision} = \frac{\boldsymbol{True\ Positive}}{\boldsymbol{True\ Positives + False\ Positives}}$$

(iii) F1 score: F1 is generally define as the average of the values obtained from recall and precision. In mathematical way f1 measure is denoted as

$$\boldsymbol{F1\ score} = \mathbf{2} * \frac{(\boldsymbol{Recall} * \boldsymbol{Precision})}{(\boldsymbol{Recall} + \boldsymbol{Precision})}$$

(iv) Cross validation: After the process of automatic categorization, one of the highly recommended method is cross validation. It is a resampling method utilized in evaluating the machine learning model on limited data.

4.2 ML Classifiers for document classification

IRF classifier: The IRF is an ensemble method utilized to construct a predictive model for the problems in classification and regression. The random trees present in the algorithm provides required output by following the ensemble method. During the time of classification problems, the decision trees do voting for standard class and the regression problems, the reply provided by the tree is approximation of the variables which are dependent to the predictors.

KNN classifier: As the same of IRF, KNN also utilized for the problems in classification and regression but widely it is used to solve problems during classification than regression. The input side consist of k-closet samples for training in feature space [20]. KNN defines a class when the points in the specific data lies on the basis of all data points.

LR classifier: It is utilized in measuring the statistical importance of each variable according to the value of probability. LR is one of the popular method to model the binomial outcome and the outcome of the LR is dichotomous in nature. It is mainly utilized in assigning the observation for the set of discrete classes. Moreover, as a classification algorithm it depends on the probability.

4.3 Classifier implementation

Initially, the data is classified into testing and training set. The size of the testing set is 25% and training set is 75%. Afterward splitting of testing and training dataset, a pipeline is used to implement the classifiers. Generally, a machine learning pipeline is used to provide better flow in the classification algorithm. The pipeline improvises the function of the complete model and helps in the step of pre-processing. The implementation of classifier using the NLP pipeline uses pickle. The pickle is utilized for object serialization and de-serialization in python. The software package is get kept in the disk and makes the work easier. It also helps in classification of new text without rewriting it for the next time.

4.4 Comparison of the classifiers

In this section, the performance measures based on precision, accuracy, F-1 score and cross validation is discussed with of the machine learning based classification algorithm such as IRF, LR, RF and KNN is computed individually and final results obtained from those classifiers are compared.

4.4.1 Precision

The results of precision obtained from the Reuters-21578 dataset on various categories such as business, entertainment, politics, sports, technology is represented in *Table.1*

Table.1 The outcome of precision parameter obtained from five categories using ML classifiers

CATEGORIES	LR (%)	KNN(%)	RF(%)	IRF(%)
Business	94	96	96.5	97
Entertainment	90	92	93	94
Politics	86	77	89	91
Sports	92	95	96	97
Technology	89	94	96	98

In the category related to business, LR obtained the precision value of 94%, KNN obtained 96%, RF obtained 96.5% and the IRF obtained 97%. In entertainment category, LR obtained the precision value of 90%, KNN obtained 92%, RF obtained 93% and the IRF obtained 94%. In political category, LR obtained the precision value of 86%, KNN obtained 77%, RF obtained 89% and the IRF obtained 91%. In the category related to sports, LR obtained the precision value of 92%, KNN obtained 95%, RF obtained 96% and the IRF obtained 97%. In technology, LR obtained the precision value of 89%, KNN obtained 94%, RF obtained 96% and the IRF obtained 98%. The graphical representation of the outcome obtained from five categories using ML classifiers is provided in the following *Figure.6*

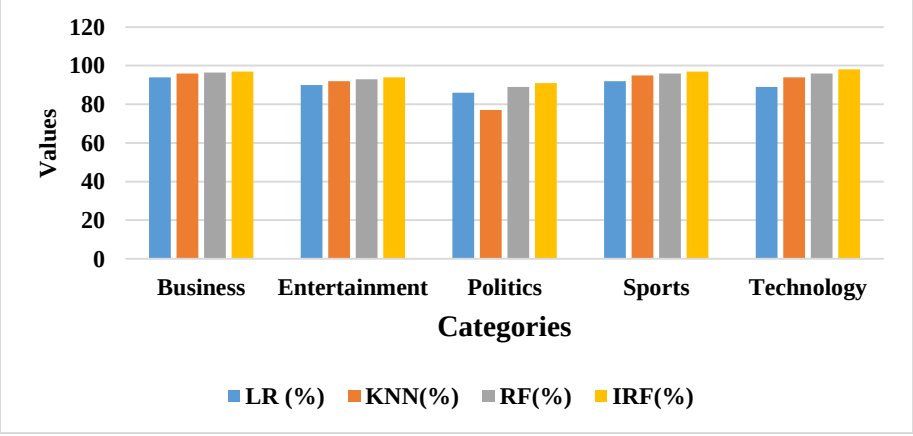


Figure 6 Graphical representation of precision value from five categories of ML classifiers

4.4.2 Accuracy

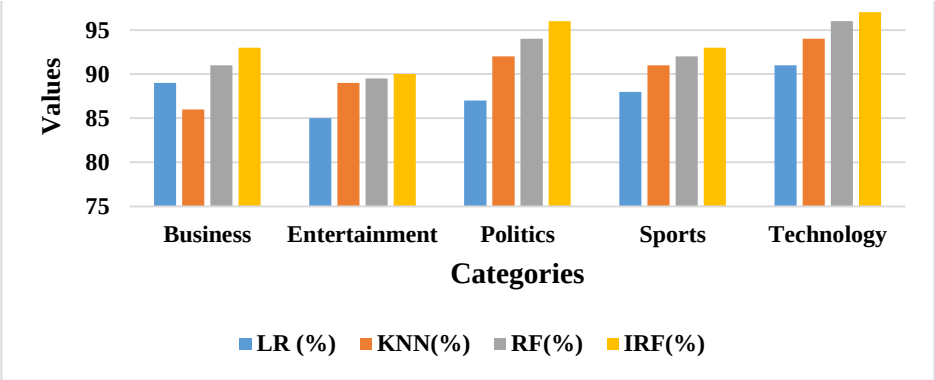
The results of accuracy obtained from the Reuters-21578 dataset on various categories such as business, entertainment, politics, sports, technology is represented in *Table.2*

Table.2 The outcome of accuracy parameter obtained from five categories using ML classifiers

CATEGORIES	LR (%)	KNN(%)	RF(%)	IRF(%)
Business	89	86	91	93
Entertainment	85	89	89.5	90
Politics	87	92	94	96
Sports	88	91	92	93
Technology	91	94	96	97

In the category related to business, LR obtained the accuracy value of 89%, KNN obtained 86%, RF obtained 91% and the IRF obtained 93%. In entertainment category, LR obtained the accuracy value of 85%, KNN obtained 89%, RF obtained 89.5% and the IRF obtained 90%. In political category, LR obtained the accuracy value of 87%, KNN obtained 92%, RF obtained 94% and the IRF obtained 96%. In the category related to sports, LR obtained the accuracy value of 88%, KNN obtained 91%, RF obtained 92% and the IRF obtained 93%. In technology, LR obtained the accuracy value of 91%, KNN obtained 94%, RF obtained 96% and the IRF obtained 97%. The graphical representation of the outcome obtained from five categories using ML classifiers is provided in the following *Figure.7*

Figure 7.Graphical representation of precision value from five categories of ML classifiers



4.4.3 F1-score

The results of F1-score obtained from the Reuters-21578 dataset on various categories such as business, entertainment, politics, sports, technology is represented in *Table.3*

Table.3 The outcome of F1-score parameter obtained from five categories using ML classifiers

CATEGORIES	LR (%)	KNN(%)	RF(%)	IRF(%)
Business	79	83	88	91
Entertainment	81	76	79	84
Politics	69	72	75	79
Sports	92	95	98	99
Technology	78	82	85	88

In the category related to business, LR obtained the F1- score value of 79%, KNN obtained 83%, RF obtained 88% and the IRF obtained 91%. In entertainment category, LR obtained the F1- score value of 81%, KNN obtained 76%, RF obtained 79% and the IRF obtained 84%. In political category, LR obtained the F1- score value of 69%, KNN obtained 72%, RF obtained 75% and the IRF obtained 79%. In the category related to sports, LR obtained the F1- score value of 92%, KNN obtained 95%, RF obtained 98% and the IRF obtained 99%. In technology, LR obtained the F1- score value of 78%, KNN obtained 82%, RF obtained 85% and the IRF obtained 88%. The graphical representation of the outcome obtained from five categories using ML classifiers is provided in the following *Figure.8*

4.4.4 Cross-validation

The results of cross-validation obtained from the Reuters-21578 dataset on various categories such as business, entertainment, politics, sports, technology is represented in *Table.4*

Figure 8. Graphical representation of F1 score from five categories of ML classifiers

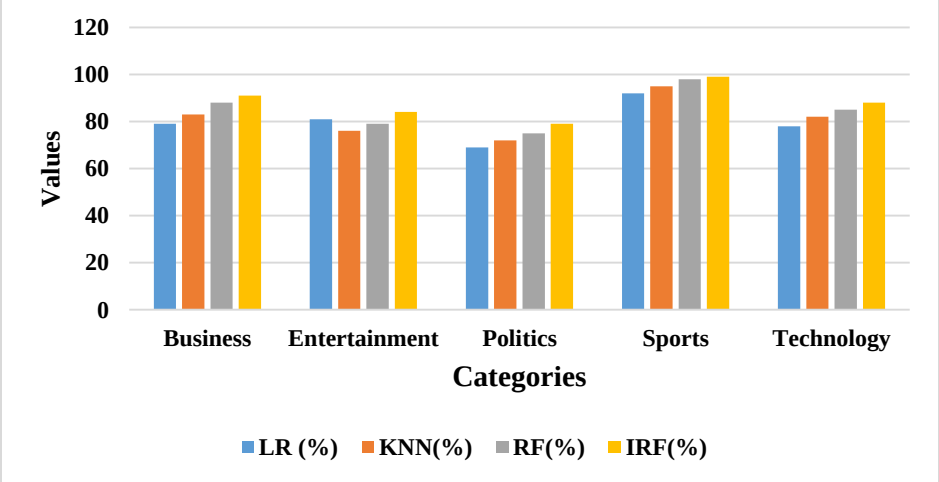


Table.4 The outcome cross validation parameter obtained from five categories using ML classifiers

CATEGORIES	LR (%)	KNN(%)	RF(%)	IRF(%)
Business	73.54	71.89	75.62	77.27
Entertainment	77	76.26	78.25	80.62
Politics	75.63	79	81	83

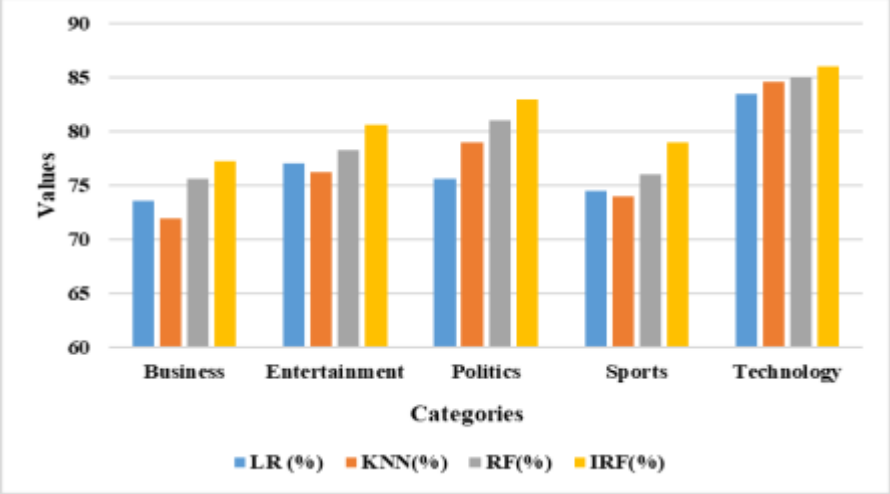
Sports	74.53	74	76	79
Technology	83.45	84.65	85	86

In the category related to business, LR obtained the cross validation value of 73.54%, KNN obtained 71.89%, RF obtained 75.62% and the IRF obtained 77.27%. In entertainment category, LR obtained the cross validation value of 77%, KNN obtained 76.26%, RF obtained 78.25% and the IRF obtained 80.62%. In political category, LR obtained the cross validation value of 75.63%, KNN obtained 79%, RF obtained 81% and the IRF obtained 83%. In the category related to sports, LR obtained the cross validation value of 74.53%, KNN obtained 74%, RF obtained 76% and the IRF obtained 79%. In technology, LR obtained the cross validation value of 83.45%, KNN obtained 84.65%, RF obtained 85% and the IRF obtained 86%. The graphical representation of the outcome obtained from five categories using ML classifiers is provided in the following *Figure.9*

4.5 Comparative Analysis

The performance of the proposed model while compared with other classifier models is mentioned in *Table.5*. The overall performance of the model is comparatively analysed with various classification models such as LSTM, LDA and SVM. The accuracy of the LSTM model is 85%, the precision is 81% and the F1-score is 82% which is comparatively lower than the proposed IRF classifier model. Since the LSTM model classifies only the offensive text and

Figure 9 Graphical representation of cross validation value from five categories of ML classifiers



doesn't classifies the text which is contained in the image document, it lacks in providing overall performance. The LDA model is compared with the proposed IRF classifier model and

the proposed model obtains 1.04% accuracy, 3.08% precision and 3.02% F1-score higher than the LDA model. Since, LDA model has difficulty in identifying the text labels that are accurate enough for classification, so the overall performance of the LDA is comparatively lower than IRF classifier model. The SVM classifier model achieves the accuracy of 76%, precision of 71%

and the F1-score of 71% which is obviously lower in all metrics because the SVM model is well suited for classifying the text documents but while coming to the classification of image documents it lacks in providing efficient performance. In overall, the proposed IRF classifier model has the capability to overcome all the drawbacks possessed by the existing methods.

Table.5 Comparative analysis

METHODS	ACCURACY (%)	PRECISION (%)	F1-SCORE (%)
LSTM model [12]	85	81	82
LDA model [13]	94	90	85
SVM classifier [16]	76	71	71
IRF classifier (proposed)	95.4	93.8	88.2

5. Conclusion

The paper presents a document classification model based on machine learning algorithm, the Improvised Random Forest (IRF) classifier. The proposed IRF classifier model is implemented on a particular dataset called Reuters-21578, a dataset consists of text based news articles. The proposed model is named as Document Classification System- Improvised Random Forest (DCS-IRF) classifier model, which efficiently classifies the text present in the documents. The classification model is implemented on the python tool kit and the obtained results are useful for the systems during retrieval of information and the proposed work helps to implement suitable method for classification of text based on precision, accuracy and F1 score parameters. The proposed model is compared with other models like LDA, LSTM and SVM classifier model. The comparison results show that DCS-IRF model achieves higher rate of accuracy, precision and F-1 measure of value 95.4%, 93.8% and 88.2% respectively. In overall, the proposed DCS-IRF system is considered as a best method for text document classification. In future, the proposed method can be implemented in classification process of image documents.

Reference

[1] Akhter, Muhammad Pervez, Zheng Jiangbin, Irfan Raza Naqvi, Mohammed Abdelmajeed, Atif Mehmood, and Muhammad Tariq Sadiq. "Document-level text classification using single-layer multisize filters convolutional neural network." IEEE Access 8 (2020): 42689-42707.

[2] Dias Canedo, Edna, and Bruno Cordeiro Mendes. "Software requirements classification using machine learning algorithms." Entropy 22, no. 9 (2020): 1057.

[3] Alhaj, Yousif A., Jianwen Xiang, Dongdong Zhao, Mohammed AA Al-Qaness, Mohamed Abd Elaziz, and Abdelghani Dahou. "A study of the effects of stemming strategies on arabic document classification." IEEE Access 7 (2019): 32664-32671.

[4] Hassanzadeh, Hamed, Mahnoosh Kholghi, Anthony Nguyen, and Kevin Chu. "Clinical document classification using labeled and unlabeled data across hospitals." In AMIA annual symposium proceedings, vol. 2018, p. 545. American Medical Informatics Association, 2018.

[5] Sahana, M., and Sandarsh Gowda MM. "NEWS CLASSIFICATION USING NATURAL LANGUAGE PROCESSING.",2022.

[6] Goodrum, Heath, Kirk Roberts, and Elmer V. Bernstam. "Automatic classification of scanned electronic health record documents." International journal of medical informatics 144 (2020): 104302.

[7] Lai, Chun-Ming, Mei-Hua Chen, Endah Kristiani, Vinod Kumar Verma, and Chao-Tung Yang. "Fake News Classification Based on Content Level Features." Applied Sciences 12, no. 3 (2022): 1116.

[8] Ali, Irfan, Nimra Mughal, Zahid Hussain Khand, Javed Ahmed, and Ghulam Mujtaba. "Resume classification system using natural language processing and machine learning techniques." Mehran University Research Journal of Engineering & Technology 41, no. 1 (2022): 65-79.

[9] Gilchrist, Cameron LM, and Yit-Heng Chooi. "Synthaser: a CD-Search enabled Python toolkit for analysing domain architecture of fungal secondary metabolite megasynth (et) ases." Fungal Biology and Biotechnology 8, no. 1 (2021): 1-19.

[10] Gupta, Vivek, Ankit Saw, Pegah Nokhiz, Harshit Gupta, and Partha Talukdar. "Improving document classification with multi-sense embeddings." arXiv preprint arXiv:1911.07918 (2019).

[11] Saleem, Zeeshan, Adi Alhudhaif, Kashif Naseer Qureshi, and Gwanggil Jeon. "Context-aware text classification system to improve the quality of text: A detailed investigation and techniques." Concurrency and Computation: Practice and Experience (2021): e6489.

[12] Wadud, Md Anwar Hussen, Muhammad Mohsin Kabir, M. F. Mridha, M. Ameer Ali, Md Abdul Hamid, and Muhammad Mostafa Monowar. "How can we manage offensive text in social media-a text classification approach using LSTM-BOOST." International Journal of Information Management Data Insights 2, no. 2 (2022): 100095.

[13] Thielmann, Anton, Christoph Weisser, Astrid Krenz, and Benjamin Säfken. "Unsupervised document classification integrating web scraping, one-class SVM and LDA topic modelling." Journal of Applied Statistics (2021): 1-18.

[14] Mora, Sara, Jacopo Attene, Roberta Gazzarata, Daniele Roberto Giacobbe, Bernd Blobel, Giustino Parruti, and Mauro Giacomini. "A NLP Pipeline for the Automatic Extraction of a Complete Microorganism's Picture from Microbiological Notes." Journal of personalized medicine 12, no. 9 (2022): 1424.

[15] Martinčić-Ipšić, Sanda, Tanja Miličić, and Ljupčo Todorovski. "The influence of feature representation of text on the performance of document classification." Applied Sciences 9, no. 4 (2019): 743.

[16] Luo, Xiaoyu. "Efficient English text classification using selected machine learning techniques." Alexandria Engineering Journal 60, no. 3 (2021): 3401-3409.

[17] Aubaid, Asmaa M., and Alok Mishra. "A rule-based approach to embedding techniques for text document classification." Applied Sciences 10, no. 11 (2020): 4009.

[18] Shah, Kanish, Henil Patel, Devanshi Sanghvi, and Manan Shah. "A comparative analysis of logistic regression, random forest and KNN models for the text classification." Augmented Human Research 5, no. 1 (2020): 1-16.

[19] Islam, Tanvirul, Ashik Iqbal Prince, Md Mehedee Zaman Khan, Md Ismail Jabiullah, and Md Tarek Habib. "An in-depth exploration of Bangla blog post classification." Bulletin of Electrical Engineering and Informatics 10, no. 2 (2021): 742-749.

[20] Aditya, B., and V. Nagaraju. "Prophecy of loan approval by comparing Decision Tree with Logistic Regression, Random Forest, KNN for better Accuracy." Journal of Pharmaceutical Negative Results (2022): 759-768.