DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008 *Int. J. Exp. Res. Rev.*, Vol. 34: 72-85 (2023)

Original Article | *Peer Reviewed* | *Open Access*

International Journal of Experimental Research and Review (IJERR)

ISSN: 2455-4855 (Online)
www.iaph.in
ISSN 2455-4855

Classification and analysis for Focused Crawled Textual Dataset for retrieving Indian origin scientists

Shivani Gautam[1*], Rajesh Bhatia[2] and Shaily Jain[3]

[1]Chitkara University School of Engineering and Technology, Chitkara University, Himachal Pradesh, India; [2]Department of Computer Science and Engineering, PEC University of Technology, Chandigarh, India; [3]Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India

E-mail/Orcid Id:

SG, shivani.gautam@chitkarauniversity.edu.in, https://orcid.org/0000-0002-7428-5155; *RB,* rbhatiapatiala@gmail.com;
SJ, shaily.jain@chitkarauniversity.edu.in, https://orcid.org/0000-0001-6078-3607

Article History:

Received: 30th Jun., 2023
Accepted: 17th Oct., 2023
Published: 30th Oct., 2023

Keywords:

Content Retrieval, focused crawler, natural language processing, Supervised Machine Learning, Text classification, web scraping

How to cite this Article:

Shivani Gautam, Rajesh Bhatia and Shaily Jain (2023). Classification and Analysis for Focused Crawled Textual Dataset for Retrieving Indian Origin Scientists. *International Journal of Experimental Research and Review*, *34*(Spl.), 72-85.
DOI : https://doi.org/10.52756/ijerr.2023.v34spl.008

Abstract: Text classification also called (text categorization or text tagging) is a crucial and extensively used approach in Natural Language Processing (NLP), to predict unseen content documents into prearranged categories. In this paper, we evaluate the dataset construction and evaluation process as a component of text classification. To begin with, we produced a newly created dataset for Indian Origin Scientists for text classification, which was collected by applying focused crawling and web scraping techniques. We then demonstrate an extensive evaluation of numerous models on this recently constructed dataset. Our evaluations display that the Random forest model outperforms the rest of the supervised models. Our results produce a fine beginning for additional research in Indian Origin Scientists' classification of text. Investigational outcome with K Nearest Neighbor, Logistic Regression, and Support Vector Machine for Indian-origin scientists produced much better performances for Random Forest when combined with SMOTE and K fold cross-validation techniques. We apply the Area under the ROC Curve to compute the effectiveness of the chosen models. Overall, the Random Forest classifier exhibited the best output along with 90% micro-average AUC.

Introduction

Classification of text is an imperative approach in natural language processing, to assign a set of prearranged categories to open-ended text data. This task is important in major applications like information retrieval, topic modeling, sentiment analysis, social media analysis, intent detection, and spam detection. A huge volume of messy unstructured data is being generated every day; text classification with machine learning can automatically structure this data to provide meaningful insights to make informed business decisions. In machine learning, classification means categorizing a data item into one or more defined classes. The data may belong to different formats like text, image, numbers, or speech. Text classification is a process of labeling the text data into one or more groups or classes. Text classification is divided into three sub-parts depending on the total number of concerned categories, for example, binary classification, multiclass classification, and multilabel classification. If there are two classes then it's called binary classification. If there are more than two classes then it's called multiclass classification whereas in multilabel classification, a document consists of one or more labels/classes attached to it. Text classification is also known as topic classification, text categorization, or document categorization. Fig 1 shows the steps in creating a text classification system.

The remainder of the paper is structured as: In section 2, we present an outline of text classification and emphasize a few latest web scraping works.

***Corresponding Author:** shivani.gautam@chitkarauniversity.edu.in

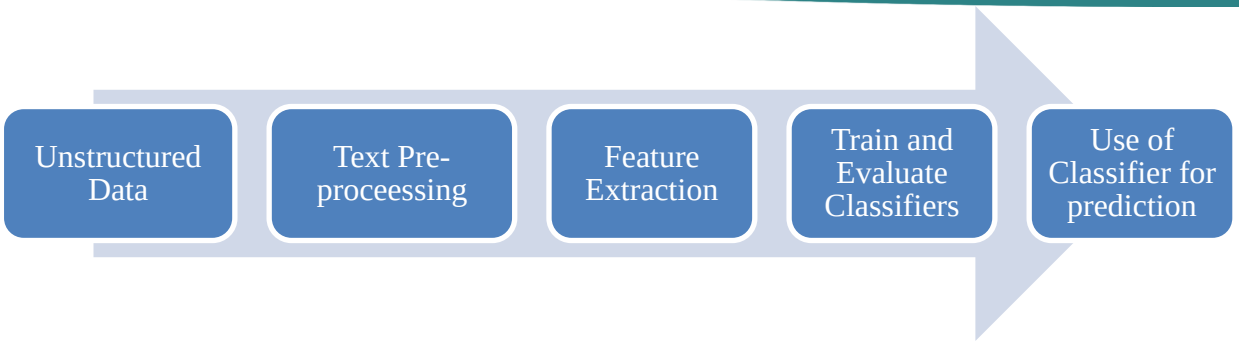


Figure 1. Text Classification Pipeline

It is accompanied by the Design and architecture in section 3, wherein we outline the dataset construction process. In section 4, various evaluation models are explained. In section 5, we cover the experiments and results. Finally, in section 6, the conclusion of our work and future scope is discussed.

Related Work

The most frequently used form of unstructured data belongs to the category of texts and speeches. It's a time-consuming and difficult task to extricate effective information from unstructured textual data. Text classification is the most commonly used NLP approach which is used to make informed business decisions in various fields. The chief aim of text classification is categorizing a class of unknown text documents, mostly with the help of supervised machine learning algorithms. Various languages like Python, Java, R, Prolog, C/C++, or MATLAB can be used for NLP tasks. Python programming language is one of the best choices for NLP as it consists of lots of packages, tools, and libraries. Natural Language Toolkit (NLTK) is a Python package that is useful for text classification for research purposes. Table 1 displays the comparison of numerous text classification techniques being used these days.

Web Scraping

Web scraping is a process that uses bots to extricate contents and data from the website in an organized manner (Glez-Peña et al., 2014). Web scraping can be of two types namely manual and automatic, manual scraping simply means copy-pasting the required data from a web page to a text file whereas automatic web scraping extracts data from a website and stores it in a structured format automatically by a bot (Saurkar et al., 2018). The web scraper is provided with the URL that needs to be scraped, then it finds specific data that needs to be extracted, the code is written and executed and the extracted data is stored in the required format. This method of extricating data from the pages can be used in various ways. For instance, it can be utilized for price comparison, social media scraping, email gathering, job listings, and research and development (Hillen, 2019). Web scraping seems to be the easy way to extract information from web pages, but it has its share of challenges. One of them is the protection policies of the website and secondly the structural changes of the website. Another challenge is to condition your web scraper as per the entries given on an individual page (Thota et al., 2021). Another shortcoming is the extraction of huge volumes of data as it would be quite time-consuming due to the detection of IP. The quality of retrieved data is another condition that may affect the scraping process (Dallmeier, 2021). There is a huge number of open-source libraries that are used in Python to extricate data. Beautiful Soup is the most commonly used library, but for JavaScript websites, Selenium can be used, which can automate browser activities as well. Scrapy is another widespread open-source web crawling structure composed in Python. It is useful for web scraping in addition to data extraction using APIs as well (Persson, 2019).

Design and Architecture (Methodology)

A list of seed URLs is prepared using the SeoQuake plugin of Google Chrome and input into the system. All the URLs present in that particular seed URL are retrieved using web scraping (beautiful soup library). For each URL in the given list, URLS are pre-processed. Keyword matching search is performed to identify relevant URLs. Then, relevant web pages are downloaded. We have used selenium with Python for the same. Relevant data is then exported to .csv. Depending on the data extraction technique, details of the scientists are extricated and kept on the database. Then data mining and refinement strategies are implemented so that the final database can be created and the search interface is prepared. Fig 2 discusses the basic design and architecture of the crawler used.

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Table 1. Comparative Analysis of Text Classification Techniques

Authors	Category	Strengths	Limitations
Kilimci et al., 2018	Deep learning &Word Embedding based Ensemble Classifier.	Improves the accuracy of classification systems by using multiple base classifiers.	Na
Patel et al., 2019	Word embeddings in keyphrase extraction.	Achieve high performance when word embeddings are associated with document-specific features, performance improvement over complex models.	In the future, integration of posterior regularization in word embedding-based CRF models can be done.
Zeng et al., 2020	An Ensemble method used for Text Classification in Clinical Trials.	Improves precision, accuracy, and recall over baseline methods.	Na
Xu, 2018	Use of Naïve Bayes classifiers for text classification.	Shows better performance than classic Naïve Bayes when Bayesian NB classifier is combined with multinomial or Gaussian event model.	Na
Kowsari et al., 2019	A survey of Text classification algorithms.	A comparison of feature extraction and recent text classification techniques has been discussed.	It is difficult to apply document categorization methods for information retrieval.
Mirończuk et al., 2018	Outline of the latest components of text classification.	Various approaches in ensemble learning like bagging, boosting, AdaBoost, stacked generalization, mixtures of experts, and voting methods were compared and discussed.	Na
Karthikeyan et al., 2019	Usage of web scraping techniques for data extraction and text classification.	The proposed model delivers a better result for the system as it gives the best results during text classification performed on extracted data with the help of effective web scraping methodologies, with a better accuracy rate.	Web scraping is a challenge due to poor response of the server and uneven transformation of data.
Pavani et al., 2017	A novel web crawling method for vertical search engines.	Proposes to retrieve hidden, relevant pages by merging rank and semantic similarity information.	The performance of the model can be improved further for relevant web pages by using various machine learning algorithms.

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Yu et al., 2018	A survey about algorithms utilized by focused web crawlers.	The advantages and disadvantages of the three crawling strategies are discussed.	To find a better combination of algorithms to enhance the crawling efficiency based on a higher harvest rate and lower computational costs.
Lunn et al., 2020	Using web scraping and natural language processing to enhance pedagogical teaching.	Discusses the effective web scraping technologies for extracting huge volumes of data from pages to create datasets.	Na
Kadhim, 2019	Survey on supervised machine learning algorithms for text classification.	Compares various supervised machine learning algorithms to organize, and extract features from the text documents.	Na
Dzisevič et al., 2019	Classification of text by utilizing various feature extraction methods.	Compares various text feature extraction methodologies on accuracy. Results show that the TF-IDF approach achieves the highest accuracy of 91% with the huge dataset.	Na
Anglin, 2019	Creating a new framework based on a gather, narrow, extract approach by utilizing web scraping and natural language processing methods.	Describes various web-scraping and text classification techniques for documents without reformatting data.	Na
Kim et al., 2019	Feature extraction using text mining for big data.	Proposes a method for extracting data using text mining for big data.	Na
Schedlbaue et al., 2021	Usage of web crawling, web scraping, and text mining for medical information market surveys.	Our study reveals how the combination of web crawling and web scraping techniques creates a dataset faster for analysis.	Na
Londo et al., 2019	Survey of Text Classification for News Articles.	Results show that the Support Vector Machine gives 93% accuracy as compared to other algorithms.	The solution needed for the imbalanced dataset for the classification work.
Onan, 2018	An ensemble approach depends on feature engineering and language function surveys for text classification.	Proposes ensemble of Random Forest with multiple features which gives the highest average predictive performance of 94.43%.	Na

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Onan, 2021	Sentiment analysis on MOOCS is dependent on text mining and deep learning methods.	Results reveal that ensemble classifiers outperform the supervised learning models as they achieve higher accuracy in educational data mining.	Na
Stein et al., 2019	Usage of using word embeddings in the analysis of hierarchical text classification.	Results indicate that the use of word embeddings is a very positive approach to hierarchical text classification.	CNN exhibited the worst effectiveness, So further investigations are required for the same.
Gupta et al., 2021	Ensemble classification is used for web page classification.	Studies propose a heterogeneous ensemble algorithm that outperforms basic models.	Web pages have information related to diverse categories which makes it difficult to classification of web pages in each category with efficiency and accuracy.
Priyadarshini, 2021	A Semantic Model used for Legal document classification by using Ensemble Methods.	Results show that the Ensemble method gives an accuracy of 98% as compared to other conventional methods.	For future scope, dynamic and live streaming of data from the websites can also be included.
Mohammed et al., 2022	An ensemble framework for classification of text.	Proposes an ensemble method that combines basic deep learning models.	Na
Deeksha et al., 2021	Web Page Classification using an Ensemble approach.	Proposes an ensemble methodology combining various basic classification models for web page classification to retrieve the Indian academician's pages from university web pages abroad.	For a reduction in the training time in the future, we can explore more groups of classifiers.
Kuriyozov et al., 2023	Uzbek language text classification dataset survey.	Reveals that Bert-based models perform better than other models and achieve the highest scores.	The model can be tested on larger datasets to further improve the performance.
Tanasescu et al., 2022	Impact of Big Data ETL Process on Text Mining study.	Shows the effectiveness of web scraping techniques for the collection and analysis of data.	Pre-trained deep learning models can be created to improve the performance.

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Landu et al., 2022	Text classification and machine learning for online News Articles.	Proposes a model for text classification using AUC as performance metrics. Also reveals a random forest model giving the best results up to 90% using the proposed model.	Na
Shrivastava et al., 2023	Usage of deep learning models for the creation of an efficient focused crawler.	Proposes an improved focused crawling approach using LSTM–CNN-based text classification model.	Na
Kaur, 2022	Usage of web scraping in sentiment analysis for news data with the help of machine learning algorithms.	Reveals how the combination of web scraping and supervised learning techniques gives better results.	Na
Muehlethaler et al., 2021	Usage of web crawling and web scraping techniques for collecting textile data from the web.	Shows how the combination of web scraping and focused web crawling techniques extracts large amounts of data in a small amount of time.	Na
Yucel et al., 2022	A new text method for classification of reviews.	Proposes a classification framework based on composite variables that outperforms all the basic models.	Na
Bajaj et al., 2023, Bajaj et al., 2022	Text classification and feature selection in flog cloud computing.	Proposes a text classification and feature selection approach for offering offloading solutions in fog cloud computing.	Na

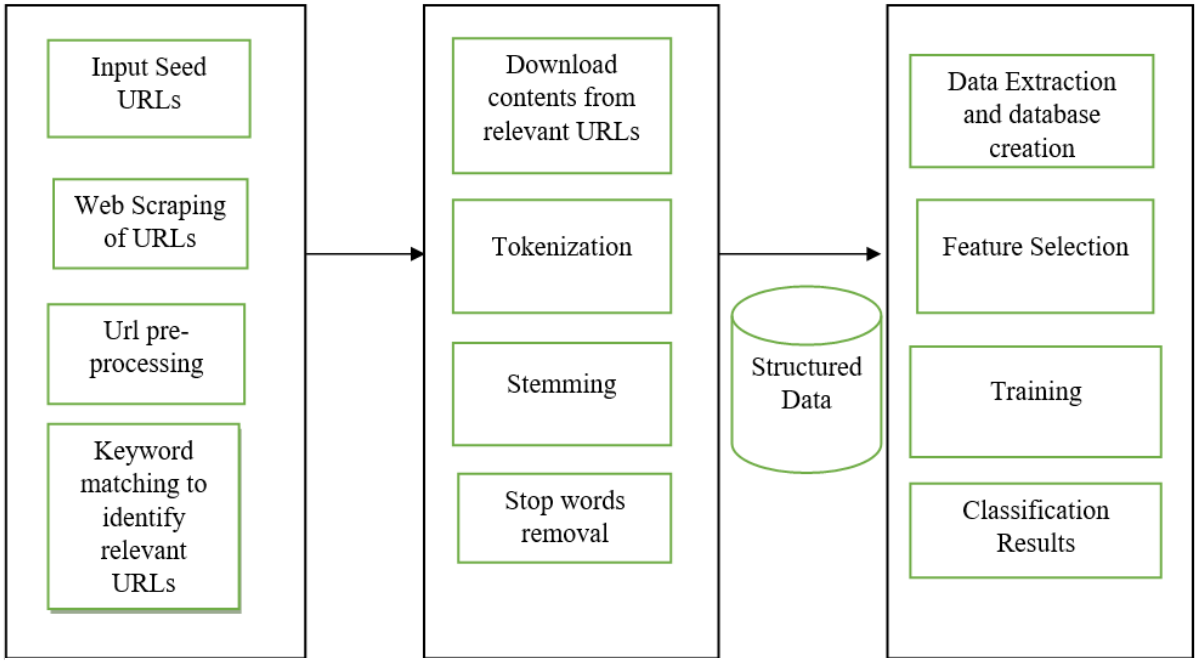


Figure 2. Pipeline of the proposed work

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

A1 | Name

	A	B	C	D	E
1	Name	Subject	Designation	Contact no.	Email
2	Kodye Abbott	IBL-S	Postdoctoral Fell	NA	kabbott@salk.edu
3	Eiman Azim	MNL-E	Assoc Professor	(858) 453-4100 xt 1074	eazim@salk.edu
4	David Acton	MNL-G	Staff Scientist	(858) 453-4100 xt 1555	dacton@salk.edu
5	David Adamowicz	LOG-G	Visiting Mentore	(858) 453-4100 xt 1009	dadamowicz@salk.edu
6	Patrick Adams	SEC	Contractor	(858) 453-4100 xt 1570	NA
7	Trinka Adamson	ARD	Sr Dir, Animal Re	(858) 453-4100 xt 1589	tadamson@salk.edu
8	Harini Adivikolanu	SNL-KT	Intern	NA	hadivikolanu@salk.edu
9	Shayan Afshar	TPEREI	Research Asst I	NA	safshar@salk.edu
10	Ravi Agarwal	LOG-G	Research Asst I	(858) 453-4100 xt 1421	ragarwal@salk.edu
11	Renata Santos	LOG-G	Research Collabc	(858) 453-4100 xt 1009	resantos@salk.edu
12	Archita Agrawal	PBL-A	Postdoctoral Fell	NA	aagrawal@salk.edu
13	Marcelo Aguilar-Riv	PBL-L	Visiting Scientist	NA	maguilarrivera@salk.edu
14	Hoda Ahmed	MCBL-W	Graduate Studen	NA	hahmed@salk.edu
15	Nasiha Ahmed	MCBL-A	Postdoctoral Fell	NA	nahmed@salk.edu
16	Christine Aiello	FIN	Sr Dir, Finance	(858) 453-4100 xt 1696	caiello@salk.edu
17	Sriram Aiyer	LOG-L	Sr Research Asso	(858) 453-4100 xt 2144	saiyer@salk.edu
18	Feras Alomireen	GRDEV	Grants Dev Appli	(858) 453-4100 xt 1507	falomireen@salk.edu
19	Dinh Albright	VCL-A	Lab Coordinator I	(858) 453-4100 xt 1016	dinhd@salk.edu

Figure 3. Overview of the final dataset retrieved

The workflow of the suggested methodology is shown in Algorithm 1 given below. It is divided into 3 stages.

Algorithm 1

Stage 1:

URL Segregation

i. The seed URL is given as input
ii. All the URLs present in that particular seed URL are retrieved using web scraping(beautiful soup library)
iii. For each URL in the given list:
iv. URLs are pre-processed (tokenization, stopwords removal, stemming)
v. If (tokens == people, staff, directory, contacts, search) then
vi. Relevant URLs
vii. else Irrelevant urls

Stage 2:

Processing of Data

- All the filtered (relevant URLs) are considered seed URLs at **this** stage.
- For each relevant URL in the list:
- Filling in the list of surnames or designations with the help of the Selenium tool(which automates the task of downloading web pages by filling in all the different surnames or designations)
- For Each web page downloaded:
- Processing is performed
  - Html tags are removed.
  - Tokenization of words is performed (splitting a large sample of text into words).
  - Stop words are removed.
  - Stemming is performed.
  - Tagging of words is performed
  - Create a list as the final data set in .csv format
- For each fetched name in .csv:
- If(fetched name or fetched _university in the dictionary of Indian surnames and Indian Universities)
- Label_data=1
- Else label_data=0
- Else Irrelevant URL

Stage 3:

Filtered Classifier

i. Load the CSV file
ii. Convert string columns to numeric using label encoding
iii. Split the dataset into features and labels
iv. Address class imbalance using SMOTE(Synthetic Minority Oversampling Technique) Oversampling
v. Split the resampled data into training and testing sets
vi. Train and evaluate different classification models using 10-fold cross-validation

The final dataset obtained through the above algorithm is being shown below in Fig 3.

Evaluation Models

In this paper, we have executed various experiments to calculate the performance of individual models on the text classification approach on Indian-origin scientists' retrieved data. The mentioned models have been utilized for research.

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Logistic Regression

Logistic regression is one of the most extensively used machine learning algorithms which is used to estimate distinct values dependent on a given group of independent variables. Its output value ranges between 1 and 0 as in spam filtering or fraud detection. In logistic regression, the algorithm helps to predict a linear relationship between the input and the output variables.

Decision Trees

A decision tree is a widespread machine learning algorithm that is used for classification as well as regression purposes. It can be represented by a binary tree which helps to estimate real values. Each node in the tree is considered an input variable *x* with a split point and each leaf in the tree consists of an output variable *y* which is used for prediction.

Support Vector Machines

SVM is a supervised machine learning model which is used for classification as well as regression. The main function of SVM is to maximize the distance between the hyperplane and the training sample dataset that is nearest to the given hyperplane. It is used for datasets having exactly two classes.

KNN

KNN belongs to the family of **supervised learning** algorithms. KNN is also called lazy learner as no learning is needed in the model. It classifies objects as per the classes of their closest neighbors in the given dataset. It takes into consideration that the more the objects are closer to each other; the more there are chances of similarities. Classification is done by a majority vote to its neighbors.

Random Forests

Random forest is an ensemble machine learning algorithm that is used for both classification and regression tasks. It is a twisted version of decision trees which consists of multiple decision trees that work together to make predictions. Each tree is being trained on an individual subgroup of the data. The concluding prediction is built by combining the predictions of all the decision trees in the forest. The greater number of trees in the forest leads to higher accuracy which in turn also prevents the issue of overfitting.

A. Addressing class imbalance using SMOTE (Synthetic Minority Oversampling Technique)

Imbalanced data can be defined as the type of dataset where the target class has disproportionate distribution of observations. In simple words, the imbalanced dataset is where the target variable has more observations in one specific class than the others. The problem of unbalanced datasets can be solved through an oversampling technique called synthetic minority oversampling (SMOTE). This algorithm creates new sample data by generating synthetic examples which is an amalgamation of the nearby minority classes. After running our dataset through SMOTE, we gathered a bigger dataset with a balanced number of classes. It also overcomes the overfitting problem which is raised by random oversampling.

B. K-Fold Cross-Validation.

Cross-validation is commonly used in machine learning algorithms for the improvement of model prediction as there is limited data to implement other better efficacy methods. If the dataset is big enough, then a test/train split can be used. But in the real world, we hardly have big enough datasets that restrict the test/train split efficacy. To solve this issue of limited data, a resampling procedure called k-fold cross-validation is used. This procedure has an exclusive variable k which defines the number of groups that a particular data sample is divided into. This technique also aids in avoiding the overfitting problem as well which happens when a model is trained with the whole of the dataset.

Finally, our goal is to view a contrast between the performances of the mentioned supervised machine learning algorithms. Furthermore, before moving towards creating the models, we need to divide the dataset between training and testing data. To implement the step, we need to initiate a sklearn function known as train_test_split and then we need to design it to reserve 80% as training data of the total dataset.

C. Performance Metrics for Evaluation

A confusion matrix is a tabular representation that determines the performance of machine learning models on a given collection of test data. The matrix exhibits 4 variables: the number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) which is produced by the framework on the test data. The matrix obtained will be a 2X2 table for binary classification.

From the confusion matrix, we can retrieve the following metrics

Accuracy: The accuracy metric is used to measure the performance of the model. It is the number of correct instances to the total number of instances.

$$\textbf{Accuracy} = \frac{\textbf{TP} + \textbf{TN}}{(\textbf{TP} + \textbf{FP} + \textbf{TN} + \textbf{FN})}$$

Precision: Precision metric is a measure of how accurate positive predictions are. It is defined as the predictions that are true to the total positive predictions.

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

Recall: Recall metric helps in measuring the effectiveness of a classification model by calculating the ratio of actual positive instances that were identified incorrectly. It is defined as the number of true positive instances to the sum of true positive and false negative instances.

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

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

Table 2. Accuracy percentage comparison with different classifiers

Algorithm	Accuracy	Precision	Recall	F1-Score
Random Forest	90%	0.94	0.87	0.90
Support Vector Machine	69.2%	0.79	0.56	0.65
K Nearest Neighbor	81.5%	0.92	0.70	0.79
Logistic Regression	60.9%	0.64	0.58	0.61
Simple Cart	81.3%	0.88	0.74	0.80
Decision Tree	81.3%	0.88	0.74	0.80

F1-Score: The F1-score metric is used to evaluate the performance of a binary classification model. It can be calculated as the harmonic mean of recall and precision.

F1 Score = 2 * Precision * Recall / (Precision + Recall)

AUC-ROC curve: AUC-ROC Curve metric is used to visualize the performance of a classification model on charts. ROC depicts a graph to display the execution of a classification model at various conception levels. The Receiver Operating Characteristic curve is drawn between two variables called True Positive Rate (TPR) and False Positive Rate (FPR) respectively. In the curve, TPR is represented on the Y-axis, whereas FPR is drawn on the X-axis. The value of AUC varies between 0 and 1. A perfect model will always have an AUC value close to 1, and therefore it will display a perfect estimate of separability.

Experiments and Results

In this paper, we have executed various experiments to calculate the performance of individual models on the text classification approach. Confusion matrices are obtained during the classification process concerning the dataset obtained for the seed URL salk.edu. In this experiment, SMOTE with cross-validation is performed using various supervised learning algorithms with ten folds. We prepared every model with the training dataset, adjusted and tweaked them by using the estimated dataset, and then tested the performance of the model by using the test dataset.

Here, we display the output of our experiments with various models used for classification of text on Indian origin scientist's dataset. We analyzed the performance of our models by utilizing specific metrics including precision, accuracy, recall, F1- score, and AUC-ROC as shown in Table 2. Based on the results of the model's performances, it can be deduced that the Random Forest model works best with 90 % accuracy. Random forest performs better than all the other models, and their performance is enhanced by adding SMOTE and k-fold cross-validation. The output of our analysis reveals the efficacy of the Random forest model for text classification on Indian-origin scientist's datasets and presents a strong groundwork for additional study in this field.

A. Comparative Analysis

The best supervised algorithm is Random Forest for classification purposes as shown through Confusion matrices and AUC-ROC curve of various classification algorithms.

Confusion Matrix

Confusion matrices are obtained during the text classification process and shown in Fig 4 concerning the salk.edu dataset.

AUC ROC Curve

In the experiment, AUC-ROC curves are obtained during the text classification process and shown in Fig 5 concerning the salk.edu dataset. It is apparent from the plots shown for various algorithms that the AUC-ROC for the Random Forest is higher than any other ROC curves. Hence, we can conclude that Random Forest works best in classifying the positive class in the dataset.

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

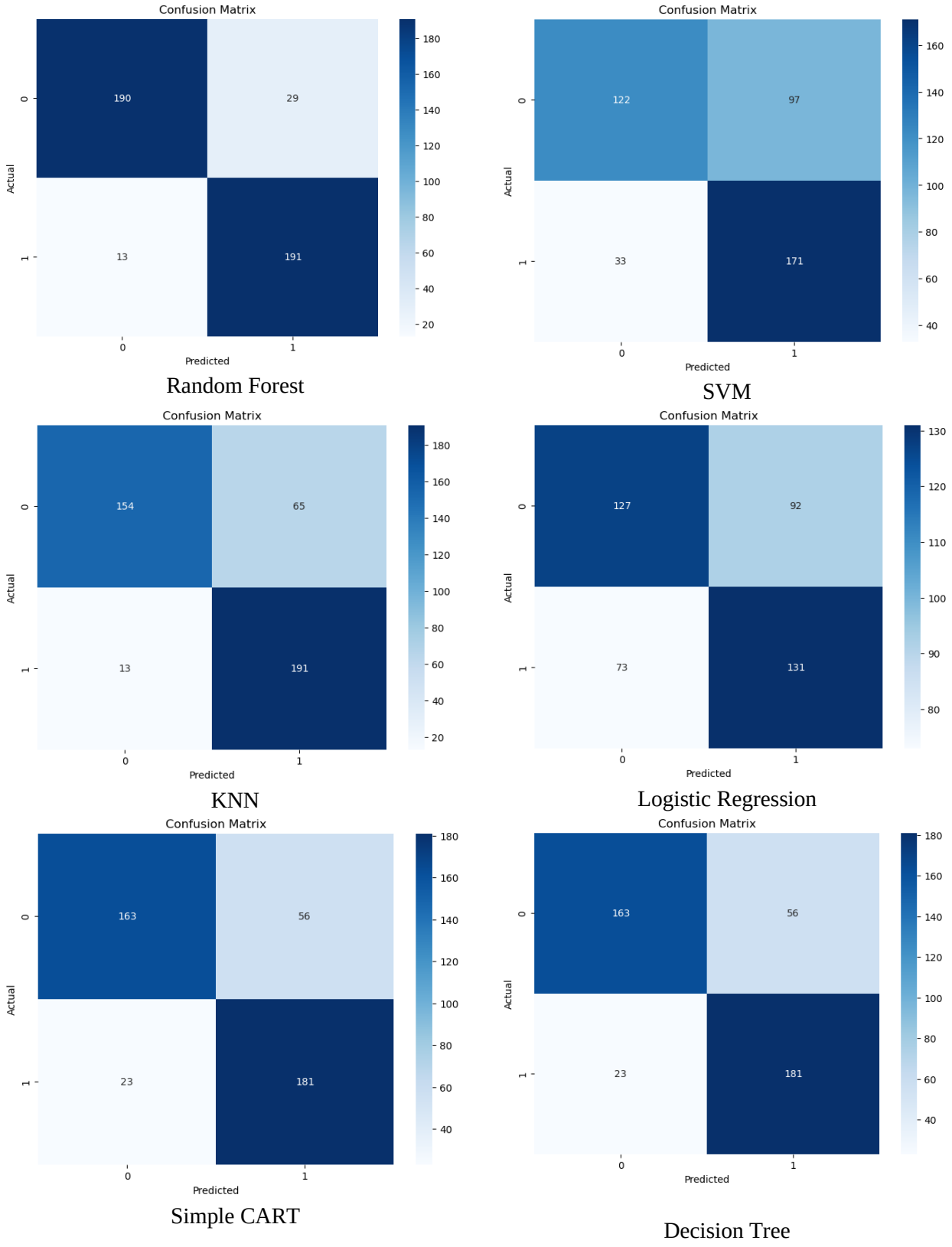
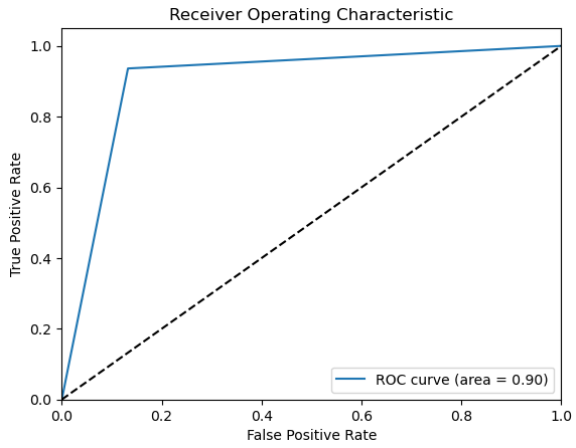
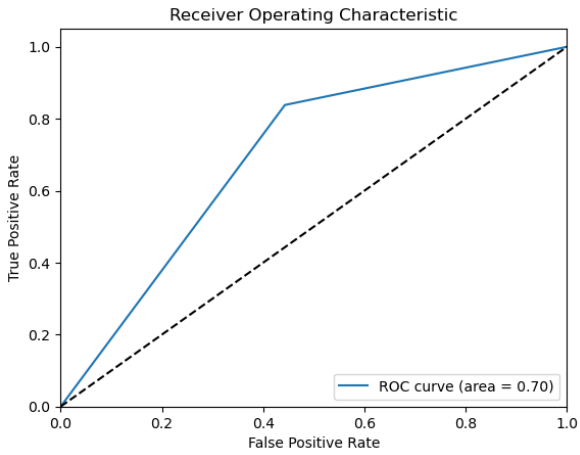


Figure 4. Matrix representation of different classifiers implementation

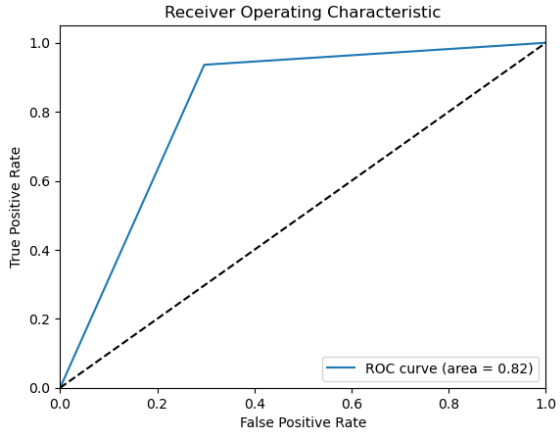
DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008



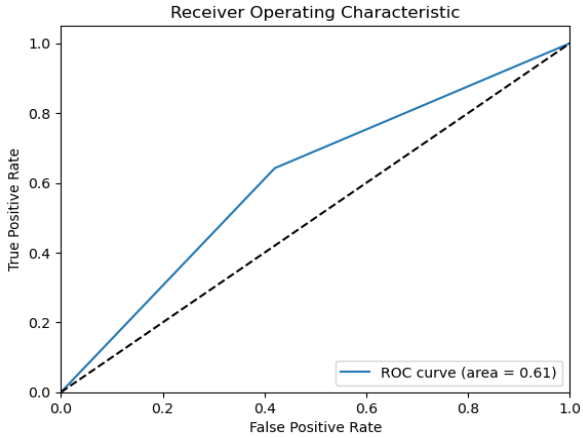
Random Forest



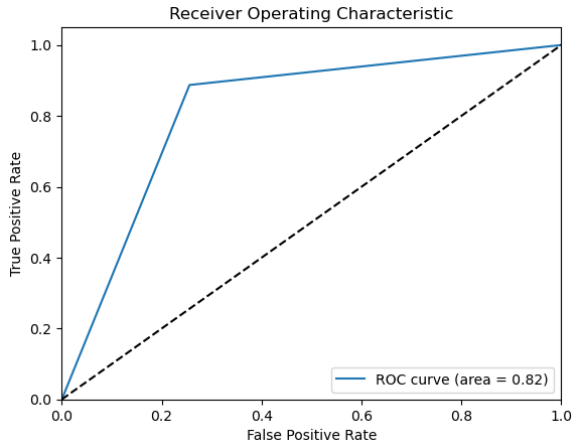
SVM



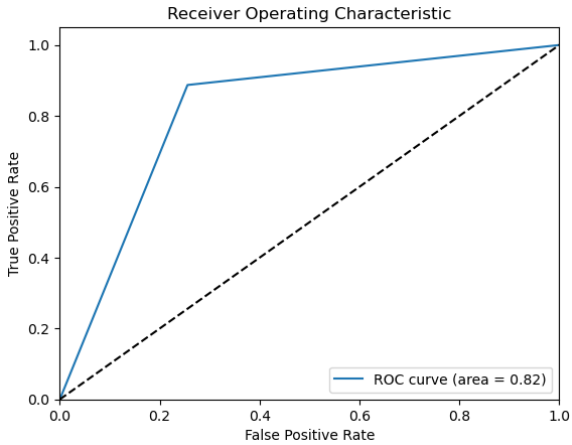
KNN



Logistic Regression



Simple CART



Decision Tree

Figure 5. AUC-ROC representation of different classifiers implementation

Conclusion and Future Work

In this study, we proposed to handle the text classification approach for the retrieval of the Indian-origin scientist dataset. Our research leads to the creation of a new dataset using focused crawling and web scraping techniques. Through the web scraping process, the unstructured data is retrieved and then converted into a structured format for additional research. The text classification task is constituted from supervised machine learning algorithms for training the model with the prepared data. In our investigations, we estimated the performance of various supervised models. Our evaluation results showed that the SMOTE with Standard Random Forest model using 10-fold cross-validation outperformed other models and achieved the highest f1-score of 90%. The output of aforesaid work shows the top performance course for text classification. In future work, we propose to enhance the performance of the

DOI: https://doi.org/10.52756/ijerr.2023.v34spl.008

models by calibrating them on a bigger dataset and to expand the research to more NLP approaches.

Conflict of Interest

The authors declare no conflict of interest.

References

Anglin, K. L. (2019). Gather-narrow-extract: A framework for studying local policy variation using web-scraping and natural language processing. *Journal of Research on Educational Effectiveness*, *12*(4), 685-706. https://doi.org/10.1080/19345747.2019.1654576

Bajaj, K., Jain, S., & Singh, R. (2023). Context-Aware Offloading for IoT Application using Fog-Cloud Computing. *International Journal of Electrical and Electronics Research*, *11*(1), 69-83. https://doi.org/10.37391/ijeer.110110

Bajaj, K., Sharma, B., Singh, R., Kumar, M., & Chowdhury, S. (2022). A comparative analysis of cloud-based services platform. *In 6th Smart Cities Symposium (SCS 2022)*, *2022*, 243-247. https://doi.org/10.1049/icp.2023.0424

Dallmeier, E. C. (2021). Computer vision-based web scraping for internet forums. IEEE, *In 2021 7th International Conference on Optimization and Applications* (ICOA), pp. 1-5. https://doi.org/10.1109/ICOA51614.2021.9442634

Deeksha, D., Bhatia, R., Bhardwaj, S., Kumar, M., Bhatia, K., & Gill, S. S. (2021). Stacking Ensemble-based Automatic Web Page Classification. *IEEE, In 2021 Fourth International Conference on Computational Intelligence and Communication Technologies (CCICT)*, pp. 169-174. https://doi.org/10.1109/CCICT53244.2021.00042

Dzisevič, R., & Šešok, D. (2019). Text classification using different feature extraction approaches. *IEEE, In 2019 Open Conference of Electrical, Electronic and Information Sciences* (eStream), pp. 1-4. https://doi.org/10.1109/eStream.2019.8732167

Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., & Fdez-Riverola, F. (2014). Web scraping technologies in an API world. *Briefings in Bioinformatics*, *15*(5), 788-797. https://doi.org/10.1093/bib/bbt026

Gupta, A., & Bhatia, R. (2021). Ensemble approach for web page classification. *Multimedia Tools and Applications*, *80*, 25219-25240. https://doi.org/10.1007/s11042-021-10891-3

Hillen, J. (2019). Web scraping for food price research. *British Food Journal*, *121*(12), 3350-3361. https://doi.org/10.1108/BFJ-02-2019-0081

Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, *52*(1), 273-292. https://doi.org/10.1007/s10462-018-09677-1

Karthikeyan, T., Sekaran, K., Ranjith, D., & Balajee, J. M. (2019). Personalized content extraction and text classification using effective web scraping techniques. *International Journal of Web Portals* (IJWP), *11*(2), 41-52. https://doi.org/10.4018/IJWP.2019070103

Kaur, P. (2022). Sentiment analysis using web scraping for live news data with machine learning algorithms. *Materials Today: Proceedings*, *65*, 3333-3341. https://doi.org/10.1016/j.matpr.2022.05.409

Kilimci, Z. H., & Akyokuş, S. (2018). Deep learning and word embedding-based heterogeneous classifier ensembles for text classification. *Complexity*, *2018*, 7130146. https://doi.org/10.1155/2018/7130146

Kim, J. C., & Chung, K. (2019). Associative feature information extraction using text mining from health big data. Wireless Personal Communications, *105*, 691-707. https://doi.org/10.1007/s11277-018-5722-5

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information*, *10*(4), 150. https://doi.org/10.3390/info10040150

Kuriyozov, E., Salaev, U., Matlatipov, S., & Matlatipov, G. (2023). Text classification dataset and analysis for the Uzbek language. arXiv preprint *arXiv*, 2302.14494. https://doi.org/10.48550/arXiv.2302.14494

Landu, T. T., Bousso, M., Loum, M. A., Sall, O., Faty, L., Dia, Y., & Sawadogo, I. (2022). Machine Learning Algorithm for Text Categorization of News Articles from Senegalese Online News Websites. In 2022, *17th Iberian Conference on Information Systems and Technologies* (CISTI), pp. 1-8. https://doi.org/10.23919/CISTI54924.2022.9820408

Londo, G. L. Y., Kartawijaya, D. H., Ivariyani, H. T., WP, Y. S. P., Rafi, A. P. M., & Ariyandi, D. (2019). A Study of Text Classification for Indonesian News Article. IEEE, *In 2019*

International Conference of Artificial Intelligence and Information Technology (ICAIIT), pp. 205-208. https://doi.org/10.1109/ICAIIT.2019.8834611

Lunn, S., Zhu, J., & Ross, M. (2020). Utilizing web scraping and natural language processing to better inform pedagogical practice. In 2020 IEEE, *Frontiers in Education Conference (FIE)*, pp. 1-9. https://doi.org/10.1109/FIE44824.2020.9274270

Mirończuk, M. M., & Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications*, *106*, 36-54. https://doi.org/10.1016/j.eswa.2018.03.058

Mohammed, A., & Kora, R. (2022). An effective ensemble deep learning framework for text classification. *Journal of King Saud University-Computer and Information Sciences*, *34*(10), 8825-8837. https://doi.org/10.1016/j.jksuci.2021.11.001

Muehlethaler, C., & Albert, R. (2021). Collecting data on textiles from the internet using web crawling and web scraping tools. *Forensic Science International*, *322*, 110753. https://doi.org/10.1016/j.forsciint.2021.110753

Onan, A. (2018). An ensemble scheme based on language function analysis and feature engineering for text genre classification. *Journal of Information Science*, *44*(1), 28-47. https://doi.org/10.1177/0165551516677911

Onan, A. (2021). Sentiment analysis on massive open online course evaluations: a text mining and deep learning approach. Computer Applications in Engineering Education, 29(3), 572-589. https://doi.org/https://doi.org/10.1002/cae.22253

Patel, K., & Caragea, C. (2019). Exploring word embeddings in crf-based keyphrase extraction from research papers. *In Proceedings of the 10th International Conference on Knowledge Capture*, pp. 37-44. https://doi.org/10.1145/3360901.3364447

Pavani, K., & Sajeev, G. P. (2017). A novel web crawling method for vertical search engines. *In 2017 International Conference on Advances in Computing, Communications and Informatics* (ICACCI), pp. 1488-1493. https://doi.org/10.1109/ICACCI.2017.8126051

Persson, E. (2019). Evaluating tools and techniques for web scraping.

Priyadarshini, R. (2021). LeDoCl: A Semantic Model for Legal Documents Classification using Ensemble Methods. *Turkish Journal of Computer and Mathematics Education* (TURCOMAT), *12*(9), 1899-1908. https://doi.org/10.17762/turcomat.v12i9.3619

Saurkar, A. V., Pathare, K. G., & Gode, S. A. (2018). An overview of web scraping techniques and tools. *International Journal on Future Revolution in Computer Science & Communication Engineering*, *4*(4), 363-367.

Schedlbauer, J., Raptis, G., & Ludwig, B. (2021). Medical informatics labor market analysis using web crawling, web scraping, and text mining. *International Journal of Medical Informatics*, *150*, 104453. https://doi.org/10.1016/j.ijmedinf.2021.104453

Shrivastava, G. K., Pateriya, R. K., & Kaushik, P. (2023). An efficient focused crawler using LSTM-CNN-based deep learning. *International Journal of System Assurance Engineering and Management*, *14*(1), 391-407. https://doi.org/10.1007/s13198-022-01808-w

Stein, R. A., Jaques, P. A., & Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. *Information Sciences*, *471*, 216-232. https://doi.org/10.1016/j.ins.2018.09.001

Tanasescu, L. G., Vines, A., Bologa, A. R., & Vaida, C. A. (2022). Big Data ETL Process and Its Impact on Text Mining Analysis for Employees' Reviews. *Applied Sciences*, *12*(15), 7509. https://doi.org/10.3390/app12157509

Thota, P., & Ramez, E. (2021). Web scraping of COVID-19 news stories to create datasets for sentiment and emotion analysis. *In The 14th Pervasive Technologies related to assistive environments conference*, pp. 306-314. https://doi.org/10.1145/3453892.3461333

Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. *Journal of Information Science*, *44*(1), 48-59. https://doi.org/10.1177/0165551516677946

Yu, Y. B., Huang, S. L., Tashi, N., Zhang, H., Lei, F., & Wu, L. Y. (2018). A survey about algorithms utilized by focused web crawlers. *Journal of Electronic Science and Technology*, *16*(2), 129-138. https://doi.org/10.11989/JEST.1674-862X.70116018

Yucel, A., Dag, A., Oztekin, A., & Carpenter, M. (2022). A novel text analytic methodology for classification of product and service reviews. *Journal of Business Research*, *151*, 287-297. https://doi.org/10.1016/j.jbusres.2022.06.062

Zeng, K., Pan, Z., Xu, Y., & Qu, Y. (2020). An ensemble learning strategy for eligibility criteria text classification for clinical trial recruitment: algorithm development and validation. *JMIR Medical Informatics*, *8*(7), e17832. https://doi.org/10.2196/17832

How to cite this Article:

Shivani Gautam, Rajesh Bhatia and Shaily Jain (2023). Classification and Analysis for Focused Crawled Textual Dataset for Retrieving Indian Origin Scientists. *International Journal of Experimental Research and Review*, *34*(Spl.), 72-85.

DOI : https://doi.org/10.52756/ijerr.2023.v34spl.008