# EFFICIENT CLASSIFICATION MODEL OF WEB NEWS DOCUMENTS WITH THE APPLICATION OF MACHINE LEARNING ALGORITHMS

**A PROJECT REPORT**

Submitted in partial fulfillment of requirement to

**RVR & JC COLLEGE OF ENGINEERING**

For the award of the degree

**B. Tech in CSE**

**by**

**MALLA SRAVANI(Y17CS099)**
**MIDDE KAVITHA(Y17CS108)**
**KONDRAGUNTA BALRAM CHOWDARY(Y17CS087)**

**JULY, 2021**

**R.V.R. & J.C. COLLEGE OF ENGINEERING**(Autonomous) (NAAC-“A+” Grade)
(Approved by **AICTE,** Affiliated to Acharya Nagarjuna University)
Chandramoulipuram::Chowdavaram,
**GUNTUR – 522 019**

## R.V.R & J.C.COLLEGE OF ENGINEERING
**(Autonomous)**

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# CERTIFICATE

This is to certify that this project work titled "**EFFICIENT CLASSIFICATION OF WEB NEWS WITH THE APPLICATION OF MACHINE LEARNING ALGORITHMS** " is The work done by M.SRAVANI (Y17CS099) , M.KAVITHA (Y17CS108), K.BALRAM CHOWDARY (Y17CS087) and submitted in partial fulfillment of the requirements for the award of degree B.Tech in Computer Science and Engineering, during the academic year 2020-2021

| | | |
|---|---|---|
| **Dr. A Sri Nagesh** | **Dr.N Venkateswara Rao** | **Dr. M. Sreelatha** |
| Professor | Professor | Prof.& Head,CSE |
| Project Guide | Project Incharge | |

# ACKNOWLEDGEMENTS

The prosperous completion of our project work , “**Efficient classification model of web news documents with the application of machine learning algorithms”** is the result of magnificent encouragement ,support and contribution of the following people.

We would like to acknowledge by expressing our gratitude to **Dr.M.Sreelatha**, head of the department of Computer Science and Engineering for her encouragement and support to accomplish this term paper.

We would like to cordially thank **Dr.N.Venkateswara Rao**, lecturer incharge and **Dr.A.Sri Nagesh**, guide for the term paper ,who played a role of motivators and facilitators, by extending their stimulation, and support to carry out this term paper successfully and also aided us in selecting the topic.

We are very much thankful to **Dr.A.Sudhakar** , Principal of RVR & JC COLLEGE OF ENGINEERING, Guntur for providing the supportive environment.

Finally we submit our sincere thanks to Lab staff in Department of Computer Science and Engineering.

**M.SRAVANI(Y17CS099)**
**M.KAVITHA(Y17CS108)**
**K.BALRAM CHOWDARY(Y17CS087)**

# Abstract

Web Applications became the popular platform for the exchange of any kind of information. This growth and increase in the data in the timely manner makes the existing methodologies to come over excessive time complexity and very high computation cost by consuming a lot of space complexity. Application of some of the data mining techniques could reduce this problem of handling this big-data. This paper proposes the proficient way of classifying the web news documents into different categories like business, entertainment, marketing, technology.

This research intends to find the efficient algorithm to automatically classify a news article in English language. First of all the news document undergo some text preprocessing method in the form of Stemming and removal of punctuation, special characters, numbers and stopwords. The main intend of performing this preprocessing step is to reduce the document noise and also the computation cost. Next we apply the feature selection onto the document to further separate important words and less important words inside the document .After applying feature selection the document the document is classified by the classifier .The comparison between different machine learning classification algorithms like Support Vector Machine(SVM), decision tree, k Nearest Neighbors(kNN) and Long short term memory(LSTM) is done .Out of all the above mentioned classifiers applied the LSTM classifier comes out with the best accuracy at 95.9% and Decision tree classifier comes out the worst at 90.2% accuracy.

# Contents

## List of Tables

# List of Figures

# List of Abbreviations

| | |
|---|---|
| LSTM | Long Short Term Memory |
| SVM | Support Vector Machine |
| DT | Decision Tree |
| kNN | k- Nearest Neighbors |
| ROC | Receiver Operating Characteristic |
| TF-IDF | Term Fequency Inverse Document Frequency |

# 1. INTRODUCTION

## 1.1 Introduction Concept

In this study it is witnessed that more applications are moving to mobile or web page- based applications instead of desktop applications. Services like shopping, billing, communication, and transportation can all be done with an internet browser or web page application instead of a sizeable application that is installed onto the desktop computers. Web page applications are commonly used to exchange information between users. However, there is a common issue with this approach, the processing of huge amounts of information on the web. Web pages for serving news have issues for providing accurate information in a timely manner since they have to serve thousands of users in short periods of time and have many articles to be processed for accuracy for easy filtering.

Despite having to handle Big-Data, web applications are the most accessible application for users to get updated information. Being able to load an application in seconds without installation attracts many users resulting in the need for huge amounts of computational power to serve users in seconds with the results they were expecting. This network traffic can be mitigated by categorizing data more affectively and classifiers can be used for data mining. The process of data mining is to find features in huge amounts of data records to minimize how much data needs to be evaluated when a user is filtering through it. These features depend on the application, with web news the features could be news content, users behavior, visiting frequency, etc. There are different branches of data mining as described in Smita,( 2014), one is the predictive branch which includes classification, regression, time series analysis, and prediction. Another is the descriptive branch which includes clustering, summarization, association rules, and sequence discovery.

Applications for data mining are to improve search engines by classifying web documents and identifying usage patterns allowing for the understanding of user behavior on a web application. The understanding of what users want to see will keep them using the application for longer and consistently.

The contribution of this article is to demonstrate how features like article content using classifiers such as Support Vector Machine (SVM), Decision Tree (DT), Long- Short Term Memory (LSTM), and K-Nearest Neighbor (k-NN) can reduce computation strain on time and space complexity. In particular, we conducted a comparative study among several machine learning algorithms for filtering web- based news contents.

The rest of this article is structured as following: section II presents related work to the study. Section III presents the proposed method using the four classifiers as mentioned above. Section IV demonstrates the obtained results after a set of experiments. Section V concludes the article with key points

### 1.2. Problem Definition

Text mining has gained quite a significant importance during the past few years. Data, now-a- days is available to users through many sources like electronic media, digital media and many more. This data is usually available in the most unstructured form and there exists a lot of ways in which this data may be converted to structured form. In many real life scenarios, it is highly desirable to classify the information in an appropriate set of categories. News contents are one of the most important factors that have influence on various sections. In this paper we have considered the problem of classification of news articles. This paper presents algorithms for category identification of news and have analyzed the shortcomings of a number of algorithm approaches.

The objective of this paper is to efficiently classify the web news into the specified four categories like health, business, entertainment and science & technology. In order to achieve this initially the Natural Language Processing techniques are applied in order to get the interesting pattern and efficient Machine Learning classification algorithms are applied like SVM, LSTM, Decision Tree, and kNN thus high accuracy is expected to be obtained.

**Significance of work**

There exists a large amount of information being stored in the electronic format. With such data it has become a necessity of such means that could interpret and analyze such data and extract such facts that could help in decision-making. Data mining which is used for extracting hidden information from huge databases is a very powerful tool that is used for this purpose. News information was not easily and quickly available until the beginning of last decade. But now news is easily accessible via content providers such as online news services. A huge amount of information exists in form of text in various diverse areas whose analysis can be beneficial in several areas. Classification is quite a challenging field in text mining as it requires prepossessing steps to convert unstructured data to structured information. With the increase in the number of news it has got difficult for users to access news of his interest which makes it a necessity to categories news so that it could be easily accessed.

Categorization refers to grouping that allows easier navigation among articles. Internet news needs to be divided into categories. This will help users to access the news of their interest in real-time without wasting any time. When it comes to news it is much difficult to classify as news are continuously appearing that need to be processed and those news could be never seen before and could fall in a new category. In this study a review of news classification based on its contents and headlines is presented.

# 2. LITERATURE REVIEW

## 2.1 Feature Selection

**Yen kan [1]** presents an approach to the segment the URL into important portions and adds components, sequential and orthographic features to model silent features. Their results show that URL based approach surpasses the performance of full content and link-based approaches content attributes classifies a web page conferring to the words and sentences it contains.

**Surajit Chandhuri [2]** The existing studies surveyed provide the overview of data warehousing and OLAP technologies, with an emphasis on their new requirements. It describe back end tools for extracting, cleaning and loading data into a data warehouse ,multidimensional data models typical of OLAP, front end client tools for querying and data analysis, server extensions for efficient query processing and tools for metadata management and for managing the warehouse.Data warehousing and on-line analytical processing (OLAP) are essential elements of decision support, which has increasingly become a focus of the database industry. Many commercial products and services are now available, and all of the principal database management system vendors now have offerings in these areas. Decision support places some rather different requirements on database technology compared to traditional on-line transaction processing applications. This paper provides an overview of data warehousing and OLAP technologies, with an emphasis on their new requirements. We describe back end tools for extracting, cleaning and loading data into a data warehouse; multidimensional data models typical of OLAP; front end client tools for querying and data analysis; server extensions for efficient query processing; and tools for metadata management and for managing the warehouse. In addition to surveying the state of the art, this paper also identifies some promising research issues, some of which are related to problems that the database research community has worked on for years, but others are only just beginning to be addressed. This overview is based on a tutorial that the authors presented at the VLDB Conference, 1996

**S.P.Deshpande[8]** represents overview of data mining system and some of its application. Information play important role in every sphere of human life. It is very important to gather data from different data sources, store and maintain the data. Generate the information. , generate knowledge and disseminate data, information and knowledge to every stakeholder. Due to vast use of computers and electronics devices and tremendous growth in computing power and storage capacity, there is explosive growth in data collection. The storing of the data in data warehouse enables entire enterprise to access a reliable current database

**Mita K. Dalal[9]** worked on text classification and feature extraction phases. Text classification can be automated successfully using machine learning techniques, however pre-processing and feature selection steps play a crucial role in the size and quality of training input given to the classifier, which in turn affects the classifier accuracy.

**Dadgar[12**] proposed an approach to classify news texts. This approach was comprised of three different steps: 1) text preprocessing, 2) feature extraction based on TF-IDF, and 3) classification based on SVM. They trained the approach through the SVM classifier which was selected because it could support data with high dimensions.

**Xindong Wu** presents the top 10 data mining algorithms .C4.5 ,k-means,SVM,Apriori ,EM,page rank,AdaBoost,kNN,Naïve Bayes and CART. These top 10 algorithms are among the most influential data mining algorithms in the research community. This paper presents the top 10 data mining algorithms identified by the IEEE International Conference on Data Mining (ICDM) in December 2006: C4.5, *k*-Means, SVM, Apriori, EM, PageRank, AdaBoost, *k*NN, Naive Bayes, and CART. These top 10 algorithms are among the most influential data mining algorithms in the research community. With each algorithm, we provide a description of the algorithm, discuss the impact of the algorithm, and review current and further research on the algorithm. These 10 algorithms cover classification, clustering, statistical learning, association analysis, and link mining, which are all among the most important topics in data mining research and development

## 2.2 Support Vector Machine:

**Thorsten Joachims[3]** introduces support vector machines for text categorization. It provides both theoretical and empirical evidence that SVMs are very well suited for text categorization.The paper considers the problem of automated categorization of web sites for systems used to block web pages that contain inappropriate content. In the paper we applied the techniques of analysis of the text, html tags, URL addresses and other information using Machine Learning and Data Mining methods. Besides that, techniques of analysis of sites that provide information in different languages are suggested. Architecture and algorithms of the system for collecting, storing and analyzing data required for classification of sites are presented. Results of experiments on analysis of web sites' correspondence to different categories are given. Evaluation of the classification quality is performed. The classification system developed as a result of this work is implemented in F-Secure mass production systems performing analysis of web content

**Hyeran Byun [4]** presented a brief introduction on SVMs and several application of SVMs in pattern recognition problems . SVMs have been successfully applied to a number of applications ranging from face detection and recognition, speaker and speech recognition,information and image retrieval, prediction and etc. because they have yielded excellent generalization performance on many statistical problems without any prior knowledge and when the dimension of input space is very high. He present a comprehensive survey on applications of Support Vector Machines (SVMs) for pattern recognition. Since SVMs show good generalization performance on many real-life data and the approach is properly motivated theoretically, it has been applied to wide range of applications. This paper describes a brief introduction of SVMs and summarizes its numerous applications.

**Chee Hong[5]** experimented an automated approach to classify online news using the SVM (Support Vector Machine) classification method. SVM has been shown to give good classification results when ample training documents are given. Classiflcation of online news, in the past, has often been done manually. In our proposed Categorizor system, we have experimented an automated approach to classify online news using the Support Vector Machine (SVM). SVM has been shown to deliver good classiflcation results when ample training documents are given. In our research, we have applied SVM to personalized classiflcation of online news. In personalized classiflcation, users can deflne their personalized categories using a few keywords. By constructing search queries using these keywords, Categorizor obtains both positive and negative training documents required for the construction of personalized classiflers. In this paper, we describe the preliminary version of Categorizor and present its system architecture.

**Daniel I Morariu[6]** investigated three approaches to build an efficient meta-classifier. In order to increase the classification accuracy. In this select 8 different SVM classifiers. For each of the classified we modified the kernel, the degree of the kernel and the input data representation. Text categorization is the problem of classifying text documents into a set of predefined classes. In this paper, we investigated three approaches to build a meta-classifier in order to increase the classification accuracy. The basic idea is to learn a meta-classifier to optimally select the best component classifier for each data point. The experimental results show that combining classifiers can significantly improve the accuracy of classification and that our meta-classification strategy gives better results than each individual classifier. For 7083 Reuters text documents we obtained a classification accuracies up to 92.04%

**D. Morariu, R. Cre¸tulescu [7]** building up on the metaclassifier presented, based on 8 SVM components, they add to these a new Bayes type classifier which leads to a significant improvement of the upper limit that the meta- classifier can reach. Krishanalal developed the intelligent News Classifier and experimented with online news from web for the category Sports, Finance and Politics**.** Text categorization is the problem of classifying text documents into a set of predefined classes. In this paper, we investigated two approaches: a) to develop a classifier for text document based on Naive Bayes Theory and b) to integrate this classifier into a meta-classifier in order to increase the classification accuracy. The basic idea is to learn a meta-classifier to optimally select the best component classifier for each data point. The experimental results show that combining classifiers can significantly improve the classification accuracy and that our improved meta-classification strategy gives better results than each individual classifier. For Reuters2000 text documents we obtained classification accuracies up to 93.87%.

### 2.3 Comparision of different classifiers:

**Yiming Yang[10]** reported a controlled study with statistical significance tests on five text categorization methods: the Support Vector Machines (SVM), a k-Nearest Neighbor (kNN) classifier, a neural network (NNet) approach, the Linear Least- squares Fit (LLSF) mapping and a Naive Bayes (NB) classifier. They focus on the robustness of these methods in dealing with a skewed category distribution, and their performance as function of the training-set category frequency .

**Rama Bharath Kumar[11]** developed stock market prediction tool. Stock market prediction is an attractive research problem to be investigated. News contents are one of the most important factors that have influence on market. Considering the news impact in analyzing the stock market behavior, leads to more precise predictions and as a result more profitable trades. So far various prototypes have been developed which consider the impact of news in stock market prediction. In this paper, the main components of such forecasting systems have been introduced. The main objective is to predict the Classify the Financial News based on the contents of relevant news articles which can be accomplished by building a prediction model which is able to classify the news as either rise or drop.

**Shri Zhong [13]** compared generative models based on the multivariate Bernoulli and multinomials distributions have been widely used for text classification. Recently,the spherical; k-means algorithm ,which has desirable properties for text clustering,has been shown to be a special case of a generative model based on a mixture of von Mises-Fisher(vMF) distributions. Text classification, document clustering and similar document analysis are the most important areas of data mining. It is currently the subject of significant global research since such areas strengthen the enterprises of web intelligence, web mining, web search engine design, and so forth. Generative models based on the multivariate Bernoulli and multinomial distributions have been widely used for text classification. Recently, the spherical k-means algorithm, which has desirable properties for text clustering, has been shown to be a special case of a generative model based on a mixture of von Mises-Fisher (vMF) distributions. This paper compares these three probabilistic models for text clustering using a general model-based clustering framework. In this work, three implementations namely, Bernoulli model-based clustering (Bernoulli-based k-means), Multinomial model-based clustering (multinomial-based k-means), von Mises-Fisher model-based clustering (vMF-based k-means) have been implemented and evaluated using suitable metrics. These algorithms have been implemented in MATLAB and evaluated with additional metric Rand Index

### 2.4 Naïve Bayes Classifier:

**Chy, Abu Nowshed[14]** has described about an approach that provides a user to find out news articles which are related to a specific classification. The naïve bayes classifier is used for classification of Bangla news article contents based on news code of IPTC. The experimental result shows the effectiveness of classification system.Web is gigantic and being constantly update. Bangla news in web are rapidly grown in the era of information age where each news site has its own different layout and categorization for grouping news. These heterogeneity of layout and categorization can not always satisfy individual user's need. Removing these heterogeneity and classifying the news articles according to user preference is a formidable task. In this paper, we propose an approach that provides a user to find out news articles which are related to a specific classification. We use our own developed web crawler to extract useful text from HTML pages of news article contents to construct a Full-Text-RSS. Each news article contents is tokenized with a modified light-weight Bangla Stemmer. In order to achieve better classification result, we remove the less significant words i.e. stop-word from the document. We apply the naive Bayes classifier for classification of Bangla news article contents based on news code of IPTC. Our experimental result shows the effectiveness of our classification.

# 3.SYSTEM ANALYSIS

## 3.1 Requirement Specification

(1) Less Response Time

(2) High data utility

(3) Reduce the effort of the user

(4) Better Performance

(5) Less computational intensity

**Requirements Model:**

Requirements analysis, also called as requirements engineering, is the process of determining user expectations for a new or modified product, These features, called requirements, must be quantifiable ,relevant and detailed. In software engineering, such requirements are often called functional specifications. Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

**Software Requirements:**

(1) Window 7/8/10,Linux operating system

(2) Chrome browser

**Hardware Requirements:**

(1) 4 GB RAM

(2) 5GB of space in hard disk

**Technology & Tools:**

(1) Python 3.7.2

(2) NLP packages

(3) Machine Learning Modules

### 3.1.1 Functional Requirements :

**Data Preprocessing**: It is extremely important that preprocess data before feeding it into our model.

**Input**: Input the dataset containing documents of the news article.

**Output**: Returns the data frame of by removing stop words and break the sentences into tokens which are stored in the pandas data frame.

**Feature Extraction:** The main idea of feature selection is to choose a subset of input variables by eliminating features with little or no predictive information.

**Input:** Input the resulted tokens from the preprocessing stage.

**Output**: Returns the feature vector which contains the words and their frequencies in all mails as key-value pairs.

**Data Classification:** Labeled records of specific category are to be used as the training data. When the classifier is trained accurately, it can be used to detect an unlabeled record.

**Input:** Input the feature vector generated from the previous step then SVM, Decision tree, kNN and LSTM Classifiers can be used to train the data.

**Output:** Return the label of the news article.

## 3.2 UML Diagrams for the Project Work

UML is an acronym that stands for Unified Modeling Language. Simply put, UML is a modern approach to modeling and documenting software. In fact, it's one of the most popular business process modeling techniques. The various UML diagrams are:

- Usecase diagram
- Activity diagram
- Sequence diagram
- Collaboration diagram
- Object diagram
- State Chart diagram
- Class diagram
- Component diagram
- Deployment diagram

### 3.2.1 Usecase Diagram:

A use case diagram is a graph of actors, a set of use cases enclosed by a system boundary, communication (participation) associations between the actors and users and generalization among use cases. In fig3.1 the actors in the usecase diagram are user and developer. The association between the usecase and actors represents the relationship. Usecases in fig3.1 are Login, Access the mails and classification. User is supposed to login and access the mails so made an association between these usecases and the user. Meanwhile we need to classify the mail as spam or ham and developer is responsible for that. So made associations between all usecases with developer.

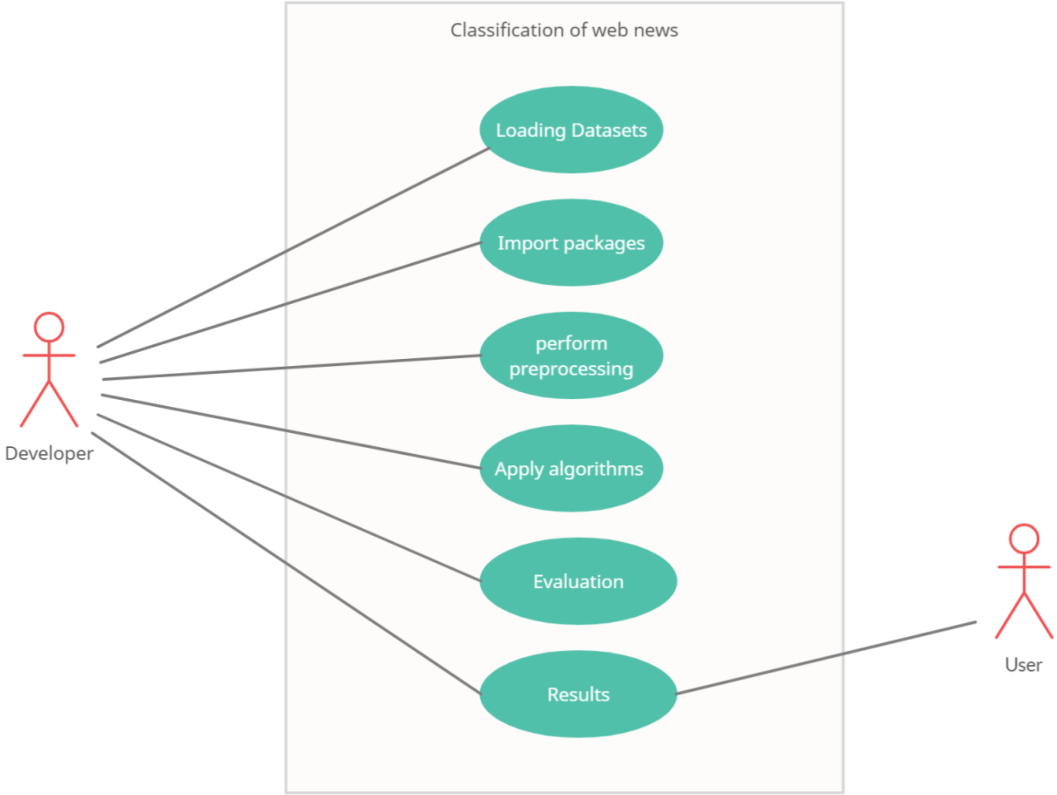


**Fig 3.2.1 Usecase diagram**

### 3.2.2 Activity diagram :

An Activity diagram is a variation of a special case of a state machine, in which the states are activities representing the performance of operations and the transitions are triggered by the completion of the operations. The purpose of Activity diagram is to provide a view of flows and what is going on inside a use case or among several classes Fig 3.2 shows the sequence of activities that are performed in order to classify the mail. Activities includes retrieving the mails from the mailbox,processing the mail,use OCR if a mail contains image attachment, Generate the feature set, Classification.

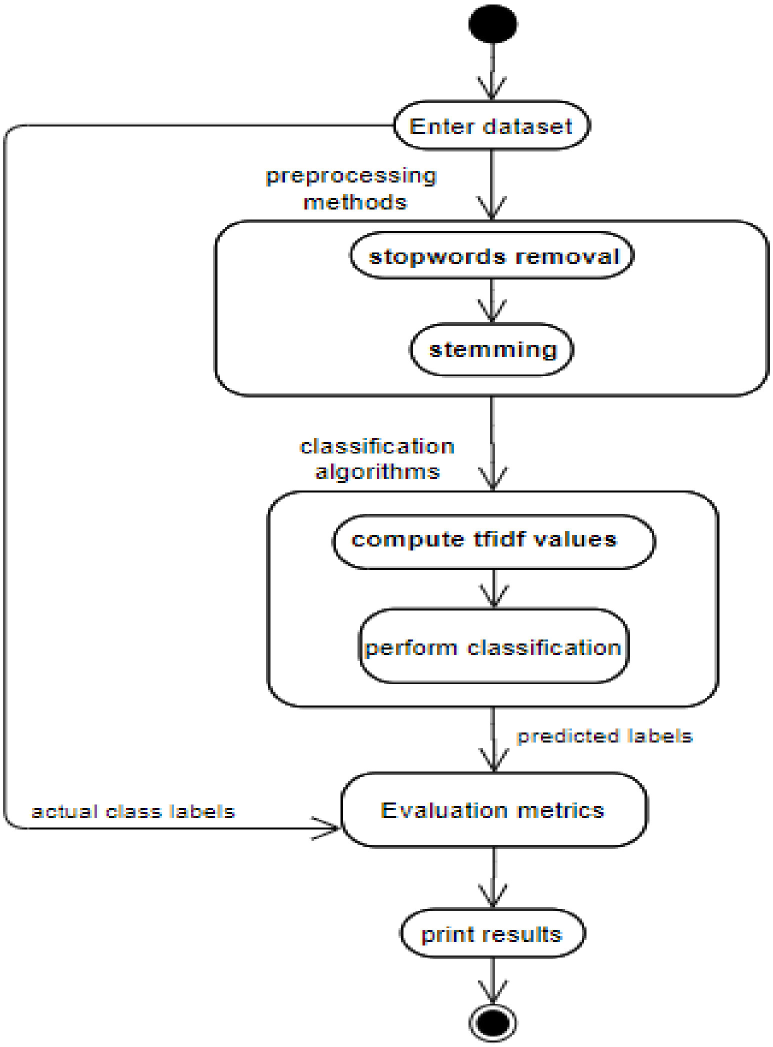


**Fig 3.2.2 Activity diagram**

### 3.2.3 Sequence diagram

A sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called as event diagrams.

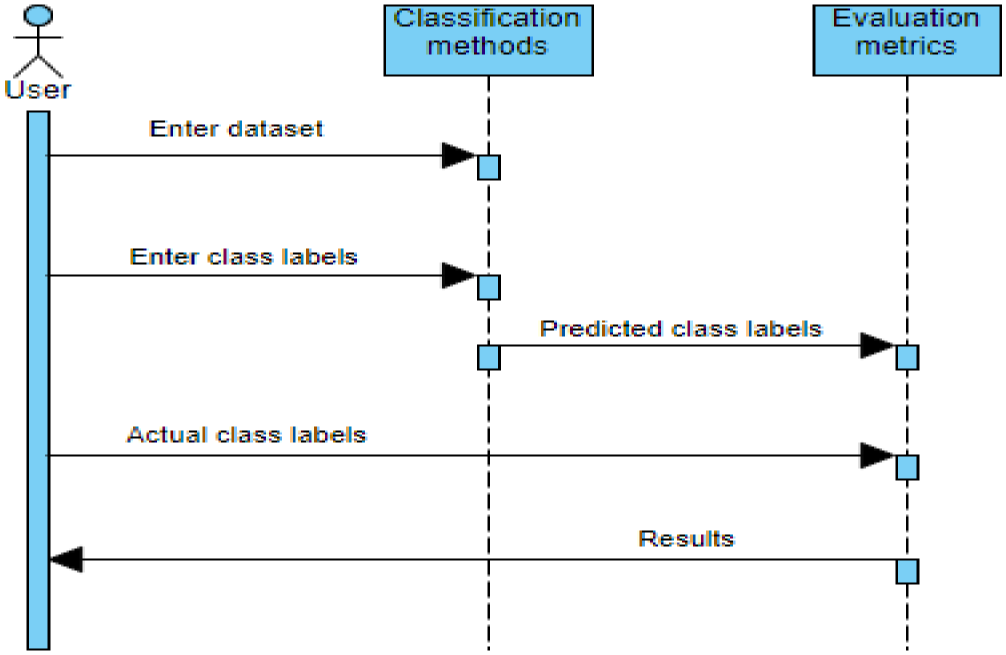


Fig 3.2.3 sequence diagram

### 3.2.4. Class Diagram

In software engineering, a class diagram in the Unified Modeling Language(UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. The class diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. In the diagram, classes are represented with boxes which contain three parts:

1. The top part contains the name of the class. It is printed in Bold, centered and the first letter capitalized.

2.The middle part contains the attributes of the class. They are left aligned and the first letter is lower case.

3.The bottom part gives the methods or operations the class can take or undertake. They are also left aligned and the first letter is lower case.

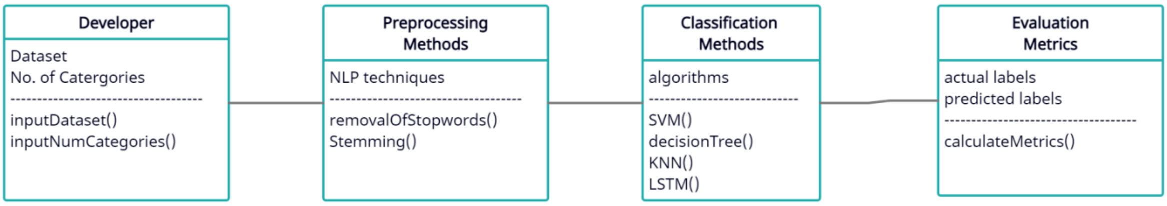


Fig 3.2.4 Class Diagram

### 3.2.5. Component Diagram

Component diagram does not describe the functionality of the system but it describes the components used to make those functionalities. So from that point component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files etc. Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment. A single component diagram cannot represent the entire system but a collection of diagrams are used to represent the whole. So the purpose of the component diagram can be summarized as:

1. Visualize the components of a system.
2. Construct executables by using forward and reverse engineering.
3. Describe the organization and relationships of the components.

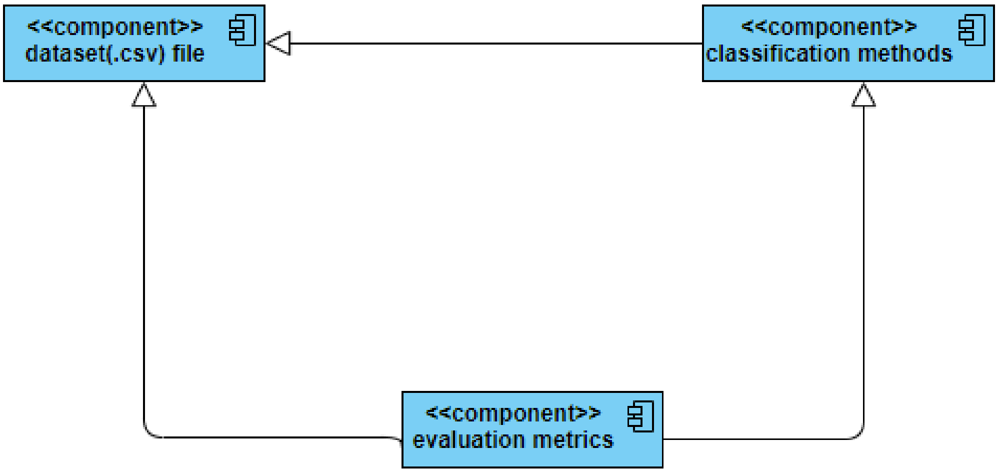


Fig 3.2.5 Component diagram

# 4.SYSTEM DESIGN

## 4.1 Architecture of the proposed method:

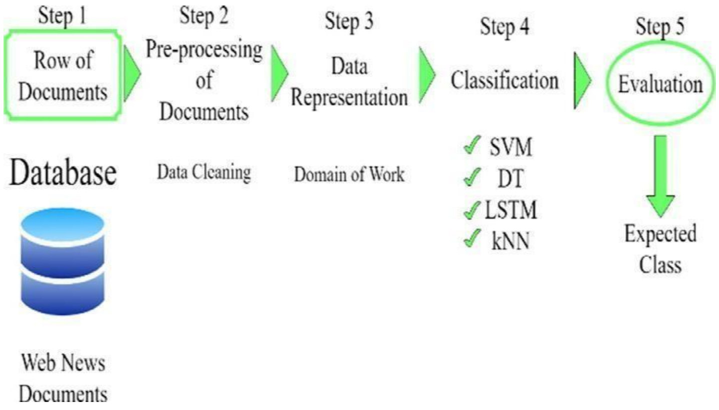


**Fig. 4.1 –Architecture of the proposed method**

The raw web news documents are collected from the available dataset of news articles.The dataset contains the many attributes like URL of the article, id of an article,story,category to which this news article belongs to and title of the news article.This web news documents are given passed through the preprocessing step.In the preprocessing step removal of punctuation,removal of stopwords ,removal of special characters,removal of numbers and removal of noisy documents is performed.

After the completion of the preprocessing phase we select the attributes that form interesting pattern for our classification model. Only the attributes which are interesting link title ,story and category are taken into consideration. After that stemming on the title attribute is performed there are many stemmers available like porter stemmer, S–stemmer etc.Porter stemmer is highly recommended. Apart from using stemming lemmatization is also suggested,where in lemmatization each and every word is reduced to its root word based on the context of the sentence.

The title attribute contains the stemmed words and frequency of the words is calculated by different count vectorizers and frequency counters and Tfidf values is generated for each and every word .That generated Tfidf values are given as an input to the classifiers namely SVM,Decision tree,kNN. As LSTM is a recurrent form of neural network the input given to the neural networks is different when compared to the other machine learning classifiers.The word embeddings and converting them from text to sequences is performed and these sequences and encoded labels are given as an input to train the recurrent neural network. The evaluation metrics like accuracy score,precision score, recall score and  F1-score are calculated and the results for each and every category are analyzed using the ROC Curve.

### 4.2 Work Flow of proposed Method:

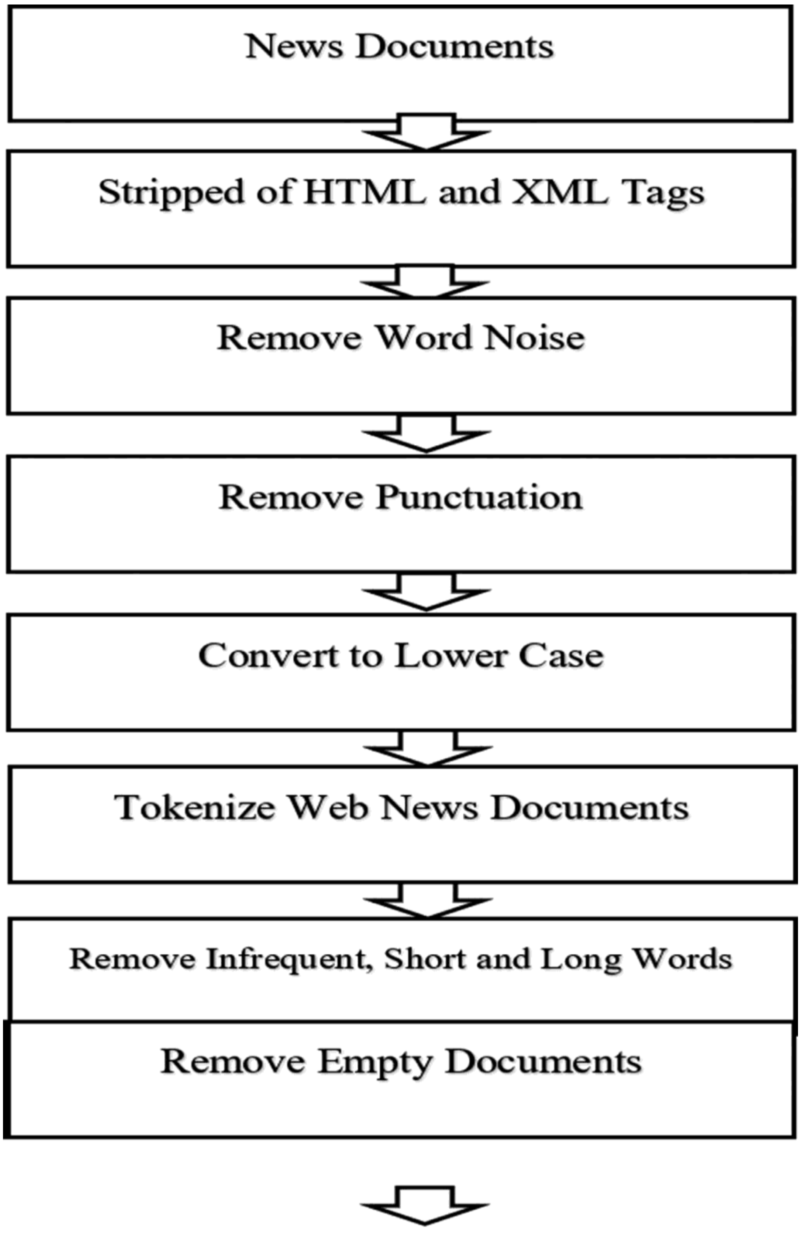

News Documents
Stripped of HTML and XML Tags
Remove Word Noise
Remove Punctuation
Convert to Lower Case
Tokenize Web News Documents
Remove Infrequent, Short and Long Words
Remove Empty Documents

**Fig: 4.2 Work flow of the preprocessing step**

## 4.3 Modular Design:

There are mainly 3 modules to be implemented, each and every module is implemented in order to classify the news article into specific category like ,it can be business, entertainment, marketing and technology. The modules are:

1. Data preprocessing
2. Data classification
3. Performance Evaluation

## 4.4 Module Description:

### 4.4.1 Data Preprocessing:

Data preprocessing is an integral step in Machine Learning as the quality of data and the useful information that can be derived from it directly affects the ability of model to learn therefore, it is extremely important that preprocess data before feeding it into our model. Three techniques to preprocess the data those are performed in order to remove unnecessary , redundant , Noisy data. They are:

- Tokenization
- Stop Word Removal
- Stemming

Preprocessing step is significant in order to make the dataset ready for classification. Documents are processed using the following steps. First, the documents are stripped of HTML and XML tags. Then, words that are considered noise are removed such as “is”, “the” and “it”. Afterword, punctuation, and special symbols are removed including “.”, “%” and “@”. The documents are then converted into lower case letters changing words like NEWS or News to news. Next, the documents are tokenized turning the articles into arrays of meaningful words so the frequency of each word can be calculated. The infrequent words that appear two or fewer times are then removed along with short words of two or fewer letters and long words. The words are then processed by Porter-Stemmer normalization algorithm Chen et al., Apr. (2018), which removes common morphological and inflexional endings from English words leaving the stem of the word for example the words “running” and “runner” would become the word “run”. After the above process, some documents will become empty, so they are removed from the array. The preprocessing steps are shown in order in Fig.4.2

**Tokenization:**

Given a character sequence and a defined document unit (blurb of texts), tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters/words, such as punctuation. Ordinarily, there are two types of tokenization:

**i) Word Tokenization**:-Used to separate words via unique space character. Depending on the application, word tokenization may also tokenize multi-word expressions like "New York". This is often times is closely tied to a process called "Named Entity Recognition".

**Example** "Machine learning makes benefit to humans" can be converted to ["Machine" ,"Learning" ,"makes" ,"benefit", "to", "humans"]

**ii)Sentence Tokenization**:-Along with word tokenization, sentence segmentation is a crucial step in text processing. This is usually performed based on punctuations such as ".", "?", "!" as they tend to mark the sentence boundaries. The raw format of email will be given as input to first step of Data preprocessing which is Tokenization. After converting the textual data to tokens. Tokens are forwarded to next step. NLTK module in python allows us to define Word and Sentence tokenization.

**Example:** "My name is Iron man. If we can't protect the world. You can be damn sure we'll avenge it." converted to

1) My name is Iron man.

2) If we can't protect the world.

3) You can be damn sure we'll avenge it.

**Stop Word Removal**:-

A stop word is a commonly used word (such as "the", "a", "an", "in") that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. We would not want these words taking up space in our database, or taking up valuable processing time. For this, we can remove them easily, by storing a list of words that you consider to be stop words. After generating the tokens in the previous step. Now we remove the stop words from the generated tokens i. e, words. Example In the list of tokens ["Machine" ,"Learning" ,"makes" ,"benefit" ,"to" ,"humans"] we remove "to" because which does not contribute to the meaning of the sentence.

**Stemming:-**

Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words known as a lemma. Stemming is important in natural language understanding (NLU) and natural language processing (NLP).

The only difference between the stem and lemma is there is no guarantee that root word gives exact meaning in stem but lemma gives the exact meaning. In the resultant list from the previous step "Learning" converted to "learn" and "benefit" to "bene". After preprocessing the data resulted root words are forwarded to feature selection step. In this work we perform some steps in Stemming:

1) A list of suffix words is stored into array with their respective root words.

2) Consider check token = availability in considered array of root word

3) If suffix of check token = true,stem the word to its respective root word from the list of array

4) Else, there is no need of stemming. Word is already in its root word format. Move to next token. Data Preprocessing might contains many steps. Based on the requirements we choose what steps needs to be done in order to get the better results. There exists many stemmers like Lovins stemmers, Porter Stemmers, Paice/Husk stemmers, Dawson stemmers, N-Gram Stemmers, HMM stemmers, YASS stemmers, Krovetz stemmers, Xerox stemmers. All the stemmers have their own pros and cons. In our research Snowball has been used which is a string processing language which creates stemming algorithms to be used for stemming purposes

**Identifying Synonym**

Using the word synonym can improve the classification process. For the synonyms detection and usage WordNet database is one of the best choice to be made. WordNet acts as a lexical database for English. It groups together English words into sets of synonyms known as synsets, it records a number of relations among these synonym sets and its members. It includes the lexical categories nouns, adjectives, adverbs, verbs but it pay no attention to prepositions, determiners, and other function words. It narrates how to find mutual informants between terms by using backdrop knowledge through WordNet.

### 4.4.2 Data Classification

Classification is the process of predicting the class of given data points. Classes are sometimes called as targets/ labels or categories. Classification predictive modeling is the task of approximating a mapping function (f) from input variables (X) to discrete output variables (y).

Web news categorization can be identified as a classification problem. This is categorical classification since there are four classes as business, entertainment, marketing, technology. A classifier utilizes some training data to understand how given input variables relate to the class. In this case, known or labeled class data have to be used as the training data. When the classifier is trained accurately, it can be used to detect an unknown email.

We used four best suited classification techniques :

- Support Vector Machine
- Decision Tree
- Long Short term Memory
- K – Nearest Neighbors

These techniques require the training dataset that is used when training the classifier and the test dataset used to test the accuracy of the classifier after training.

**Support Vector Machine:**

SVMs work by graphing the dataset where the number of features equals the dimensions of the graph. The dataset points are then separated by hyper planes which in the 2D senses with two classes consist of three parallel lines, two of these lines are known as the support vectors they border the closest data point(s) between classes. The last line marks the mid- point between the two support vectors. The hyper plane with the greatest distance between the two support vectors is the best fit. If there isn’t one the dataset must be scaled to the next dimension using a kernel function. Popular kernel functions include polynomial Eq(4.1) ,gaussian as in Eq(2) radial basis function (RBF), Laplace RBF, sigmoid, etc.

$$k\left(x_i, x_j\right) = \left(x_i x_j + 1\right)^d \tag{4.1}$$

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right) \tag{4.2}$$

### Decision Tree

A DT builds a flow chart diagram that looks like a tree and branches off at every decision or variable. The topmost node of the tree is known as the root node, the tree‟s creation originates at the root node and is built in a top-down fashion. Entropy shown in Eq. (4.3), is used to measure how unpredictable a decision is within the tree. 1 being very unpredictable, a 50% chance and 0 being a guarantee, either a 0% or 100% chance were $p_i$ is the probability of a class i and c is the total classes. Information gain is used to measure the reduction of uncertainly when additional nodes are used before the given node, Eq. (4.4). The set of nodes that provide the most information gain are used. Decision trees are prone to overfitting since this above process is repeated on every branch of the tree.

$$E(s) = \sum_{i=1}^{c} -p_i \log_2 p_i \tag{4.3}$$

$$IG(X, Y) = E(Y) - E(Y|X) \tag{4.4}$$

### Long Short-Term Memory

LSTM specializes in text classification since the classifier can learn long-term dependencies between the text. The LSTM classifier is a form of recurrent neural network or RNN, which is a layered network that uses the previous outputs for the inputs of the next layer. LSTM has feedback connections al- lowing it to work with sequences of data instead of just single data points. An LSTM node consists of a cell, input gate, output gate, and forget gate. The cell is what remembers values over a time interval and the three gates regulate how the information will flow through the cell. The following Eqs. (4.5)–(4.10) are used in the creation of an LSTM with a forget gate. Where W and U are matrices containing the weights for the inputs and recurrent connections. $x_t$ is the input vector unit, $f_t$ is the for- get activation vector, $i_t$ is the input activation vector, $o_t$ is the output activation vector, $h_t$ is the output vector unit, $\tilde{c}_t$ is the cell input activation vector and $c_t$ is the cell state vector. $\Sigma_g$ and $\sigma_h$ are the activation functions sigmoid and hyperbolic tangent, respectively.

$$f_t = \sigma_g\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{4.5}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{4.6}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{4.7}$$

$$\tilde{c}_t = \sigma_h(W_c x_t + U_c h_{t-1} + b_c) \tag{4.8}$$

$$c_t = f_t \circ c_{i-1} + i_t \circ \tilde{c}_t \tag{4.9}$$

$$h_t = o_t \circ \sigma_h(c_t) \tag{4.10}$$

**K Nearest Neighbors**

The kNN technique works by using documents from the training dataset after they are preprocessed to build its model. The model uses the Euclidian distance equation shown in Eq. (4.11). The Euclidian distance is calculated between a new document from the test dataset that was also preprocessed and all of the training data points.

$$d_j = \sqrt{\sum_{i=1}^{n} (WN_c - WN_o)^2} \tag{4.11}$$

where $d_j$ is the Euclidian distance between the current document ($WN_c$) and another document ($WN_o$). After all the Euclidian distances are calculated for the new news document they are sorted from the smallest distances to the largest. The k nearest data points are selected based on the distance and the mode class in the k points is used as the prediction. The model is recreated many times with different values for k to see which k will produce the fewest errors when new data is introduced to the model.

# 5. Implementation

## 5.1 Algorithms:

The algorithms are used to cluster the time series datasets that are used in this work. Four classification algorithms are used in this experiment.

### 5.1.1. Support Vector Machine:

Support vector machine was implemented using MATLAB R2020a's fitcecoc function. The SVM achieved an accuracy of 95.47% making SVM the leading classification technique for the dataset. Since SVMs have no trouble handling large feature datasets.

The step wise algorithm is presented for an individual document below:-

**Step1**: Consider a random record from the newsCorpus record which contains the attributes of the news article

**Step2**: The considered news article is in raw form. To perform the feature extraction/selection and classification procedure, initially email is needed to pre-process. Pre-processing involves the steps of tokenization, stemming and stop word removal.

**2.1.** Initially, tokenize the email into individual keywords. Tokenization split each individual word into different token.

**2.2.** Remove the stop words from the obtained tokens.

**2.3** .Remove the punctuation,Special characters and numbers

**2.4.** Perform stemming on the tokens obtained from the previous step. Stemming process reduces the size of word to its root word. For stemming, a predefined list of possible words with their respective stem words is considered.

**2.4.1.** For stemming, a list of suffix words is stored into array with their respective root words.

**2.4.2.** Consider check token = availability in considered array of root word

**2.4.3.** If suffix of check token = true, stem the word to its respective root word from the list of array.

**2.4.4.** Else, there is no need of stemming. Word is already in its root word format. Move to next token.

**Step 3:** Apply Tf-idf vectorizer to each and every word such that for every unique word in our document a vector of float type is assigned. Now these Tf-idf values of each token is given as input to the SVM classifier.

**Step 4:** Based on the evaluated feature similarity using SVM classification model of tokens they are labeled according to the target categories list which contains target labels business, entertainment, technology and marketing.

**Step 5:** Store the record that was classified and repeat the process for all the records.

### 5.1.2 Decision Tree Technique:

Decision Tree was implemented using the MATLAB fitctree function. DT achieved an accuracy of 90.72% making DT the second-best solution. A decision tree is a flowchart-like tree structure where an internal node represents feature(or attribute), the branch represents a decision rule, and each leaf node represents the outcome. The topmost node in a decision tree is known as the root node. It learns to partition on the basis of the attribute value. It partitions the tree in recursively manner call recursive partitioning. This flowchart-like structure helps you in decision making. It's visualization like a flowchart diagram which easily mimics the human level thinking. That is why decision trees are easy to understand and interpret**.**

The step wise algorithm is presented for an individual document below:-

**Step1**: Consider a random record from the newsCorpus record which contains the attributes of the news article

**Step2**: The considered news article is in raw form. To perform the feature extraction/selection and classification procedure, initially email is needed to pre-process. Pre-processing involves the steps of tokenization, stemming and stop word removal.

**2.1.** Initially, tokenize the email into individual keywords. Tokenization split each individual word into different token.

**2.2.** Remove the stop words from the obtained tokens.

**2.3** .Remove the punctuation,Special characters and numbers

**2.4.** Perform stemming on the tokens obtained from the previous step. Stemming process reduces the size of word to its root word. For stemming, a predefined list of possible words with their respective stem words is considered.

**2.4.1.** For stemming, a list of suffix words is stored into array with their respective root words.

**2.4.2.** Consider check token = availability in considered array of root word

**2.4.3.** If suffix of check token = true, stem the word to its respective root word from the list of array.

**2.4.4.** Else, there is no need of stemming. Word is already in its root word format. Move to next token.

**Step 3:** Apply Tf-idf vectorizer to each and every word such that for every unique word in our document a vector of float type is assigned. Now these Tf-idf values of each token is given as input to the Decision tree classifier.

**Step 4:** Based on the evaluated feature similarity using Decision tree classification model of tokens they are labeled according to the target categories list which contains target labels business, entertainment, technology and marketing.

**Step 5:** Store the record that was classified and repeat the process for all the records.

### 5.1.3 k-Nearest Neighbors classifier:

The kNN technique works by using documents from the trainng dataset after they are preprocessed to build its model. kNN algorithm assumes the similarity between the new case/data and available cases category that is most similar to the available categories.kNN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using kNN algorithm.

The step wise algorithm is presented for an individual document below:-

**Step1**: Consider a random record from the newsCorpus record which contains the attributes of the news article

**Step2**: The considered news article is in raw form. To perform the feature extraction/selection and classification procedure, initially email is needed to pre-process. Pre-processing involves the steps of tokenization, stemming and stop word removal.

**2.1.** Initially, tokenize the email into individual keywords. Tokenization split each individual word into different token.

**2.2.** Remove the stop words from the obtained tokens.

**2.3** .Remove the punctuation,Special characters and numbers

**2.4.** Perform stemming on the tokens obtained from the previous step. Stemming process reduces the size of word to its root word. For stemming, a predefined list of possible words with their respective stem words is considered.

**2.4.1.** For stemming, a list of suffix words is stored into array with their respective root words.

**2.4.2.** Consider check token = availability in considered array of root word

**2.4.3.** If suffix of check token = true, stem the word to its respective root word from the list of array.

**2.4.4.** Else, there is no need of stemming. Word is already in its root word format. Move to next token.

**Step 3:** Apply Tf-idf vectorizer to each and every word such that for every unique word in our document a vector of float type is assigned. Now these Tf-idf values of each token is given as input to the kNN classifier by setting the kneighbors attribute as 5.

**Step 4:** Based on the evaluated feature similarity using kNN tree classification model of tokens they are labeled according to the target categories list which contains target labels business, entertainment, technology and marketing.

**Step 5:** Store the record that was classified and repeat the process for all the records.

### 5.1.4 Long Short Term Memory

The long short-term memory classifier was implemented with the MATLAB trainNetwork function. LSTM achieved an accuracy of 95.93 % making LSTM the first best solution. Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. This is a behavior required in complex problem domains like machine translation, speech recognition, and more.LSTMs are a complex area of deep learning. It can be hard to get your hands around what LSTMs are, and how terms like bidirectional and sequence-to-sequence relate to the field.

The step wise algorithm is presented for an individual document below:-

**Step 1:** Consider a random record from the newsCorpus record which contains the attributes of the news article

**Step2**: The considered news article is in raw form. To perform the feature extraction/selection and classification procedure, initially email is needed to pre-process. Pre-processing involves the steps of tokenization, stemming and stop word removal.

**2.1.** Initially, tokenize the email into individual keywords. Tokenization split each individual word into different token.

**2.2.** Remove the stop words from the obtained tokens.

**2.3** .Remove the punctuation, Special characters and numbers

**2.4.** Perform stemming on the tokens obtained from the previous step. Stemming process reduces the size of word to its root word. For stemming, a predefined list of possible words with their respective stem words is considered.

**2.4.1.** For stemming, a list of suffix words is stored into array with their respective root words.

**2.4.2.** Consider check token = availability in considered array of root word

**2.4.3.** If suffix of check token = true, stem the word to its respective root word from the list of array.

**2.4.4.** Else, there is no need of stemming. Word is already in its root word format. Move to next token.

**Step3:** The text is converted into the sequences ,inorder to decrease the overhead of unequal lengths in the sequences padding of the sequences is done.The target variables list are encoded using the one-hot encoding model.

**Step4:** The available sequences and encoded target variables are given for recurrent neural network to perform the training phase.

**Step5:** Based on the evaluated feature similarity using LSTM recurrent neural network model of tokens they are labeled according to the target categories list which contains target labels business, entertainment, technology and marketing.

**Step6:** Store the record that was classified and repeat the process for all the records.

### 5.2 Datasets

The Artificial Intelligence Lab at the Faculty of Engineering ,Roma Tre University Italy originally provided the dataset Gasparetti(2017).The dataset contains 422,937 news articles in four different categories, entertainment, science and technology, business and health.For the experiment 25,000 news articles were randomly chosen from all the categories and for classification purposes divided into two subsets. The training dataset which included 90% or 95% of the random dataset and the test dataset containing the remaining data in the random dataset.

The fields that are present in the dataset are website URL, Content of the web news,author of the article,date at which the web news is published, serial number these five attributes are separated by "/" the data is to be extracted neatly and kept in an organized way in a text documents.The dataset contains the data about the news articles with the above mentioned attributes. The classes are eventually spread throughout the dataset.The distribution of the classes in the dataset is shown in the fig (5.2.1)

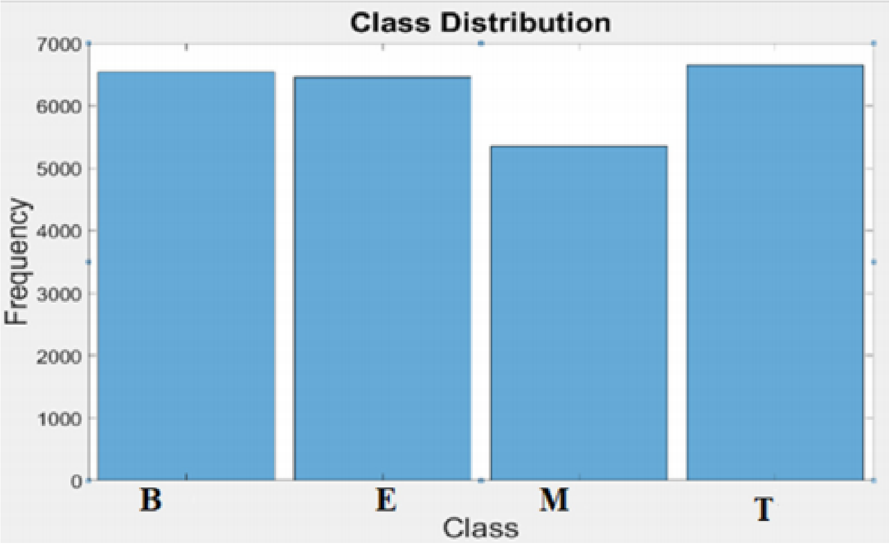


**Fig : 5.2.1 Class distribution in the dataset**

## 5.3 Metrics

The performance measure provides the efficiency, accuracy and reliability of a training model. Many factors that contributes to the performance of classifier and model. Some of those we consider four parameters precision,recall,f1-score and accuracy for SVM, Decision tree, kNN and LSTM

**Metrics for Classification:**

**Precision:-**

It can be defined as the probability of predicted target value with true value using the classifier. It also defines the effectiveness of classifier. It can be formulated as mentioned below

$$\text{Precision } P=TP/(TP+FP) \qquad (5.3.1)$$

**Recall:-**

It can be defines as the probability of actual label of the class. It can be formulated as mentioned below.

$$\text{Recall } R=TP/(TP+FN) \qquad (5.3.2)$$

**F1-Score:-**

It is a measure to define the overall performance of the classifier. It is evaluated from the precision and recall values as mentioned in below.

$$\text{F1-Score } F= 2*P*R/(P+R) \qquad (5.3.3)$$

**Accuracy:-**

It is the ratio of number of correct predictions to the total number of input samples.The equation for the accuracy is shown as Eq(5.3.4)

**Sensitivity:-**

Sensitivity is a measure of the proportion of actual positive cases that got predicted as positive.The equation for sensitivity is shown as Eq(5.3.5)

**Specificity:-**

Specificity is defined as the proportion of actual negatives, which got predicted as the negative.The equation for specificity is shown as Eq(5.3.6)

$$accuracy = \frac{\sum(Y_{pre} = Y_{test})}{Y_{test}} * 100 \tag{5.3.4}$$

$$sensitivity = \frac{TP}{TP + FN} \tag{5.3.5}$$

$$speci\,ficity = \frac{TN}{TN + FP} \tag{5.3.6}$$

In this experiment, the confusion matrix is used to demonstrate the preferred result against the predicted result. Each column of the matrix represents samples in the predicted class while each row represents samples in the actual class. Receiver operating characteristics (ROC) curves are also used throughout the article to demonstrate a true positive rate against the false positive rate at various thresholds. The dataset is being organized into four classes instead of two so in order to use ROC curves the one verse all approach is used. The area under a ROC curve (AUC) and accuracy of a classification technique are good indicators for comparison.

# 6.TESTING

## 6.1 Testing done:

### Introduction to testing:

Testing is a fault detection technique that tries to create failure and erroneous states in a planned way. This allows the developer to detect failures in the system before it is released to the customer. Note that this definition of testing implies that a successful test is test that identifies faults. We will use this definition throughout the definition phase. Another often used definition of testing is that it demonstrates that faults are not present.

Testing can be done in two ways:

1.Top down approach

2.Bottom up approach

### 1.Top down approach:

This type of testing starts from upper level modules. Since the detailed activities usually performed in the lower level routines are not provided stubs are written.

### 2.Bottom up Approach:

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embedded within the larger system. In this project, bottom up approach is used where the lower level modules are tested first and the next ones having much data in them.

### Testing Methodologies

The following are the Testing Methodologies:

- Unit Testing
- Integration Testing.
- User Acceptance Testing.
- Output Testing.
- Validation Testing

This concept is experimented with the newscorpus dataset in which 25000 data records are chosen randomely from the dataset.

## 6.2 Test Cases

**Test Case 1:-** In this test, we upload a valid .csv data file and print the dataset

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 1 | Verify Data set | Uploading valid Dataset | Valid dataset | Upload respective .csv data file | No error | Dataset is printed | Pass |

Table (6.1) Test case 1

**Test Case 2:-** In this test, we upload an invalid data file other than .csv data file and print the dataset .

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 2 | Verify Data set | Uploading Dataset | Invalid dataset | Upload another file | Has to show error | Invalid format exception | Fail |

Table(6.2) Test Case 2

**Data preprocessing:-**

**Test Case 3:-**In this test, we test whether the stopwords are removed from the data or not

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 3 | Preprocessing | Remove the stop words | Uploaded dataset of news article | Import stopwaords from NLTK | Has to show no errors | Shows the test without stopwords | Pass |

Table(6.3) Test case 3

**Feature Extraction:-**

**Test Case 4:-** In this test,the feature vector generation is done

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 4 | Feature vector generation | Using Count vectorization and TFIDF transformer | Uploaded dataset of news article | Import tfidf transformer | Has to convert each and every word to vector | Converted each and every word to vector form of float value | Pass |

Table(6.4) Test case 4

**Test Case 5**:- In this test, testing whether the classifier is correctly classified or not is checked

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 5 | Perform classification | Perform the SVM classification | The feature vectors that are generated for each and every article | Using SVM | Get the label of the news article | Returns the correct label of the mail | Pass |

Table(6.5) Test Case 5

**Test Case 6:-**

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 6 | Perform classification | Perform the SVM classification | The feature vectors that are generated for each and every article | Using SVM | Get the label of the news article | Returns the wrong label of the mail | Fail |

Table (6.6) Test Case 6

**Test Case 7:-** In this test, we test the initialize_K nearest neighbors function of kNN by supplying valid dataset and  k value.

| Test case Id | Test scenario | Test Case | Test data | Test steps | Expected result | Actual result | Status |
|---|---|---|---|---|---|---|---|
| 7 | Perform classification | Initializing random values for k | Dataset and valid number of neighbors(k) | Execute initializing random number of neighbors function | Selection of random number of neighbors | K nearest neighbors are selected | Pass |

Table(6.7) Test Case 7

# 7.Results

## 7.1 Actual Results

**Table 7.1.1 Results of different classifiers**

| ALGORITHM | Accuracy (percentage ) | Precision(in percentage) | Recall(in percentage) | F1-Score(in percentage) |
|---|---|---|---|---|
| LSTM | 95.9 | 95.9 | 95.9 | 95,9 |
| SVM | 95.3 | 95.3 | 95.3 | 95.3 |
| kNN | 94.7 | 94.7 | 94.7 | 94.7 |
| DT | 90.2 | 90.2 | 90.2 | 90.2 |

Out of all the classifiers implemented LSTM coming out with the best accuracy of 95.9 % and decision tree coming out the least at 90.2% of accuracy.

## 7.2 Analysis of the Results obtained

**Confusion Matrix:**

A **confusion matrix** is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

**ROC Curve:**

An **ROC curve** (**receiver operating characteristic curve**) is a **graph** showing the performance of a classification model at all classification thresholds.

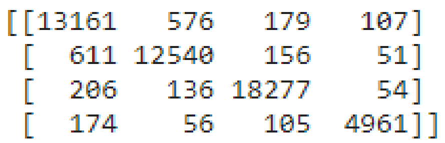
This **curve** plots two parameters:

1.True Positive Rate

2. False Positive Rate

### 7.2.1 Analysis of Results for SVM

Confusion Matrix of SVM:



```
[[13161   576   179   107]
 [  611 12540   156    51]
 [  206   136 18277    54]
 [  174    56   105  4961]]
```

Fig: 7.2.1.1 Confusion matrix for SVM

ROC Curve for SVM:

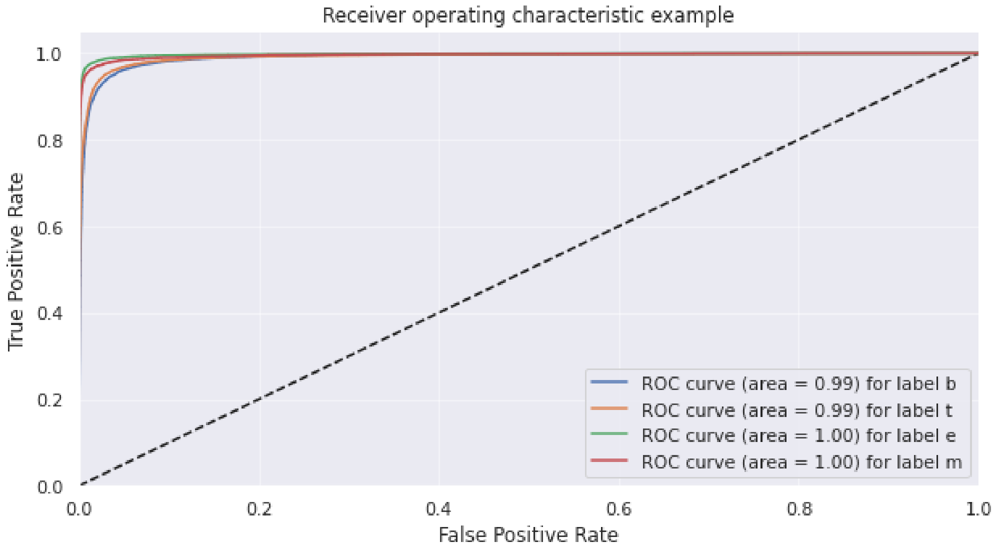
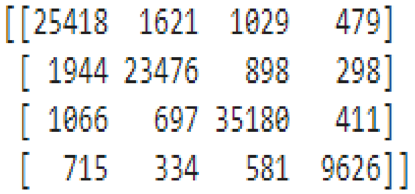


Fig: 7.2.1.2 ROC Curve for SVM

### 7.2.2 Analysis of Results for Decision Tree classifier

Confusion Matrix for Decision Tree classifier



```
[[25418  1621  1029   479]
 [ 1944 23476   898   298]
 [ 1066   697 35180   411]
 [  715   334   581  9626]]
```

7.2.2.1 Confusion Matrix for Decision Tree Classifier

ROC Curve for Decision Tree:

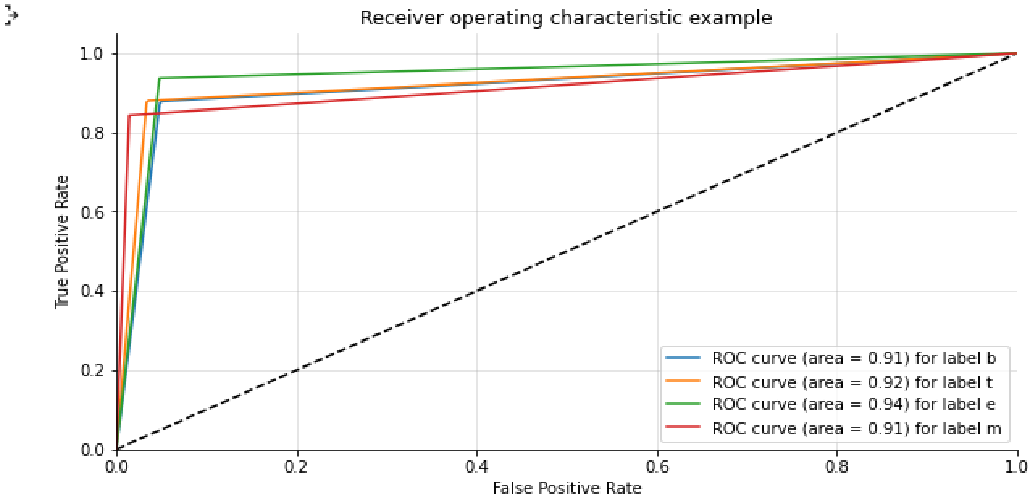
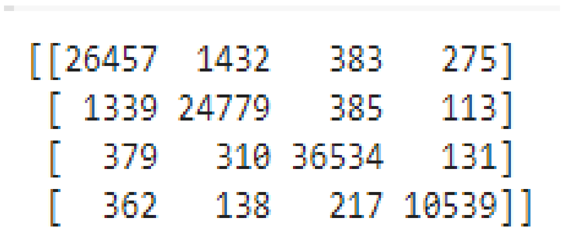


Fig 7.2.2.2 ROC Curve for Decision Tree Classifier

### 7.2.3 Analysis of Results for kNN Classifier

Confusion Matrix for kNN Classifier:



```
[[26457  1432   383   275]
 [ 1339 24779   385   113]
 [  379   310 36534   131]
 [  362   138   217 10539]]
```

Fig :7.2.3.1 Confusion Matrix for kNN Classifier

ROC Curve for kNN Classifier:

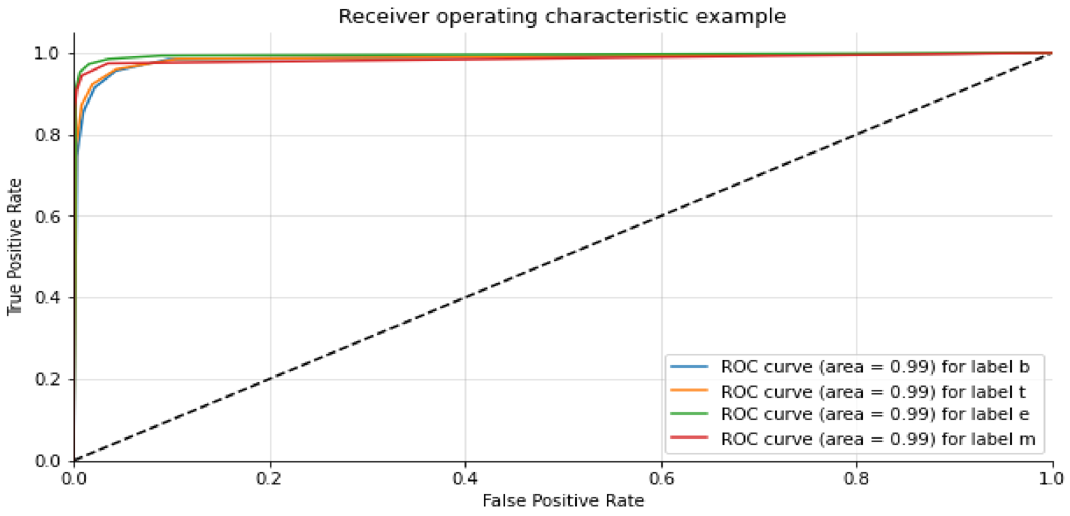


Fig: 7.2.3.2 ROC Curve for kNN Classifier

### 7.2.4  Analysis of Results for LSTM Classifier

ROC Curve for LSTM Classifier:

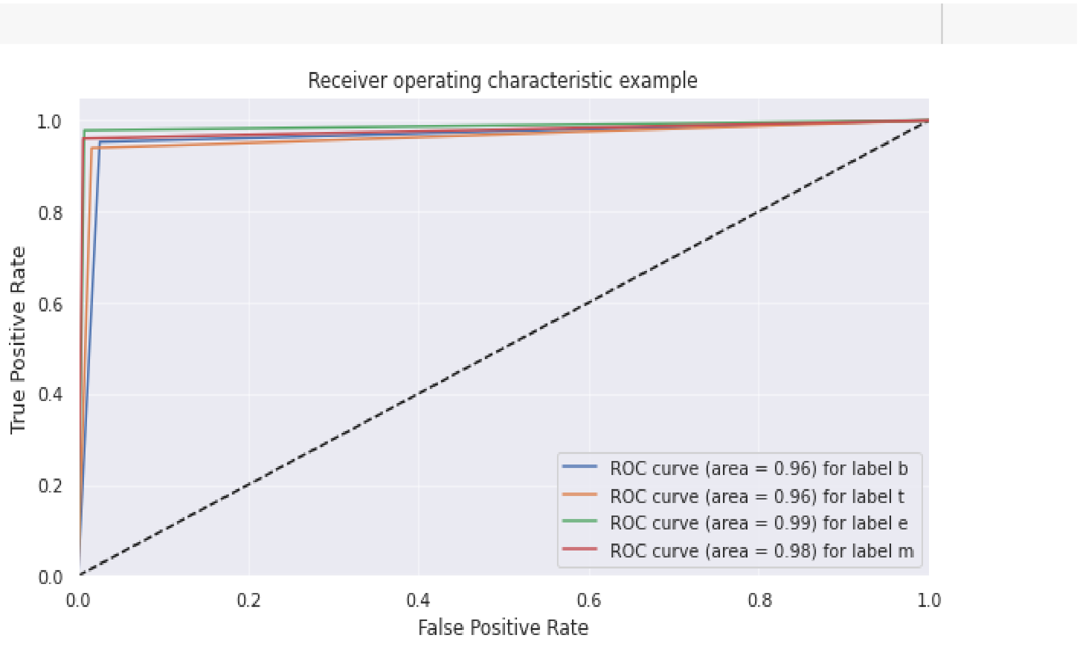


Fig: 7.2.4.1 ROC Curve for LSTM Classifier

## 8. CONCLUSION AND FUTURE WORK

After recognizing the daily impact, the web applications have on users and their popularity, related studies were discussed to find the research gap and different suitable machine learning classifiers are studied. We studied the machine learning classifiers that demonstrated a more accurate results with less time and space complexity for web applications based big data. Specifically, we have compared four classifiers K-Nearest Neighbors (kNN), Support Vector Machine (SVM), Decision Tree (DT), and Long Short-Term Memory (LSTM). LSTM coming out on top with the best accuracy of 95.9% and Decision Tree with the worst at 90.2%. It is important to note that the web mining in combination with classification has become a valued research topic over recent years.

Existing solutions fulfill the needs of most web applications. However, existing solutions require a huge amount of computation power, so the proposed approach works on reducing space and time complexity. The results of the top classifiers demonstrated good accuracy, a reduction in time needed for the training and testing phases of classification (time complexity), and reduction of total documents (space complexity).

# 6.REFERENCES

[1]https://www.academia.edu/2038312/Fast_webpage_classification_using_features

[2] Surajit Chandhuri, Umeshwar Dyal,“ An overview of data warehousing and OLAP technologies” Published in ACM Sigmod record, Vol. 26, Issue 1, pp. 65-74, USA, 1997

[3] Thorsten Joachims,“Text Categorization With Support Vector Machines: Learning with many relevant features,” Published in: European Conference on Machine Learning (ECML 1998), Lecture notes in Computer Science Book series Springer Publications, Vol. 1398, pp. 137-139, 1998

[4] Hyeran Byun Pattern Recognition with Support Vector Machines, First International Workshop, SVM 2002, Niagara Falls, Canada, August 10, 2002, Proceedings

[5] Chee-Hong Chan Aixin Sun Ee-Peng Lim,“Automated Online News Classification with Personalization,” 4th International Conference on Asian Digital Libraries (ICADL2001), Bangalore, pp. 320-329, 2001

[6] Daniel I. Morariu, Lucian N. Vintan, Volker Tresp, “MetaClassification using SVM Classifiers for Text Documents”, World Academy of Science, Engineering and Technology 21, pp. 15-20, 2006

[7]D. Morariu, R. Cre¸tulescu, L. Vin¸tan,“Improving a SVM Meta-classifier for Text Documents by using Naïve-Bayes,” Int. J. of Computers, Communications & Control, Vol. 3, pp. 351-361, 2010.

[8] Mr. S.P Deshpande, Dr. V.M Thakre,"Data mining system and Applications: A review,” In International Journal of Distributed and Parallel System (IJDPS), Issue 1, Vol. 1, pp. 32-44, 2010

[9] Mita K. Dalal, Mukesh A.Zaveri,“Automatic text classification: A technical review,” In International Journal of Computer Applications, Vol. 28, Issue 2, pp. 37-40, 2011.

[10] Yiming Yang, Xin Liu,"A re-examination of text categorization methods,” Caenegie Mellon University Pittsburg, PA 15213- 3702, USA, Vol. 18, Issue 2, 2012.

[11] Rama Bharath Kumar, Bangari Shravan Kumar, Chandragiri Shiva Sai Prasad,“Financial news classification using SVM,” International Journal of Scientific and Research Publications, Vol. 2, Issue 3, pp. 1-6, 2012

[12] S. M. H. Dadgar, M. S. Araghi, M. M. Farahani,"A novel text mining approach based on TF-IDF and Support Vector Machine for news classification," 2nd IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, pp. 112-116, 2016

[13]**https://www.researchgate.net/publication/289272630_Performance_evaluation_of_three_model-based_documents_clustering_algorithms**

[14]https://www.researchgate.net/publication/280560028_Bangla_news_classification_using_naive_Bayes_classifier