International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319-6734, ISSN (Print): 2319-6726
www.ijesi.org ||Volume 9 Issue 9 Series I || September 2020 || PP 21-33

Comparative Study on Text Classification

Amina Khatun, Md. Mafiul Hasan Matin, Md. Al-Amin Miah, Md. Robbani Miah
[1](Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh)
[2](Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh)
[3](Department of Computer Science and Engineering, University of Chittagong, Bangladesh)
[4](Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh)

ABSTRACT: *Text classification also known as text categorization, which is one of the important field in natural language processing and the task of automatically sorting a set of documents into categories from a predefined set. It has many applications in the commercial world like automated indexing of scientific articles according to predefined thesauri of technical terms, filing patents into patent directories, selective dissemination of information to information consumers, automated population of hierarchical catalogues of Web resources, spam filtering, identification of document genre, authorship attribution, survey coding, and even automated essay grading. Automated text classification is attractive because it frees organizations from the need of manually organizing document bases, which can be too expensive, or simply not feasible given the time constraints of the application or the number of documents involved. There are many algorithms used in text classification where few of them are essential. In this thesis, we implemented four categorization engines based on Naive Bayes, Decision Tree, Support Vector Machine and K-Nearest Neighbor methodologies. We then compared the effectiveness of these four engines by calculating standard precision and recall for a collection of documents. We will further report the time efficiency of these four engines.*

KEYWORDS –*Categorization, SVM,Naive Bayes, Decision Tree, KNN, DSR, TP, TN, FP, FN, ML, TF-IDF*

Date of Submission: 20-08-2020 Date of Acceptance: 05-09-2020

I. INTRODUCTION

There are vast amounts of information that are available from the online text documents and lots of knowledge that are concealed within these documents. When the correct intelligent tools are determined and applied it to these documents, the knowledge can be easily find-out. Machine learning (ML) works with the design and development of algorithms and techniques that approve the system to learn from data to flourish the performance of system [1]. The problem that conformed the almost researchers in the field of machine learning is how to select the most appropriate model/algorithm to be use on a given application [2]. Data mining is one of the most important machine learning application [4,3]. Data mining is the process which analysis the vast amounts of data that are stored in the computers and discovered patterns in it. Machine learning and data mining techniques are using to automatically find out patterns and classify from the documents [5]. Data mining decrease the costs in time and money, so that it becomes most popular in the fields of science, analysis and healthcare.Text classification is one of the weightiest research issues and common technique in the data mining.Text categorization is the automatic classification of text documents under predefined categories or classes. The main problem is how to make a classification model i.e. make a classifier to classify the text document into one or more class. Information Retrieval (IR) and Machine Learning (ML) techniques are used to assign keywords to the documents and classify them into distinct categories. Machine learning aid us to categorize the documents automatically. Information Retrieval aid us to represent the text as an attribute [19]. The procedure initially begins with the parsed documents and important terms gained from the training documents after preprocessing techniques. These documents are employed to train the categorizer. Once the training phase is done, categorization engines based on Naïve Bayes, Decision Tree, Support Vector Machine and K–Nearest Neighbor are implemented to predict the categories of the documents. We then compare the effectiveness of these four engines by calculating standard precision and recall for test documents. A paper titled "Toward Optimal Feature Selection in Naive Bayes for Text Categorization" by Bo Tang,Haibo He, Steven Kay was published on IEEE: Feb 2016.In this paper, Automated feature selection is important for text categorization to reduce the feature size and to speed up the learning process of the classifiers. Accuracy is very low based on wordcount [6]. A paper titled "A Bayesian Classification Approach Using Class-Specific Features for Text Categorization" by Bo Tang, Haibo He, Paul M.Baggenstoss, Steven Kay was published IEEE:2016. In this paper, they present a Bayesian classification approach for automatic text categorization using class-specific features [8]. "Comparative study of classification algorithm for text-based categorization" by O Ardhapure, G Patil, D Udani, K Jetha. This paper,reveals that text categorization is a process in data mining which assigns predefined categories to new text documents using machine learning algorithms.Any document in the form of text,image,music etc. can be classified using some categorization technique.Accuracy is not compared with much data [10]. A paper titled “Text classification by combining text classifiers to improve the efficiency of classification” by Aaditya Jain,Jyoti Mandowara.Basic working of web crawler is presented in this paper.Only working of text classification is given but nothing is given about how pages can be ranked using some algorithms [12]. "A Comparative study on text categorization" by Karamcheti, Aditya Chainulu.In this paper they performed text classification on a small dataset by applying only two machine learning algorithm Naïve Bayes and K-Nearest Neighbor [11]. The accuracy of Naïve Bayes and K-NN is not commendable for large amount of data.

II. BACKGROUND AND LITERATURE REVIEW

Classification is a form of data analysis that extracts models describing important data classes [32]. It is the most popular and widely used machine learning technique. Such models which are known as classifier can predict categorical class labels under supervision learning. The predictions are discrete and unordered. No intermediate value can be obtained from classifier. For example, a classifier is built to detect whether an image contains the picture of a dog or a cat. The prediction will be either “dog” or “cat”. No intermediate value can be obtained from classifier. Classification learning technique can be used on labeled data. There are two types of data in classification learning. One type is called training data, another is called test data. Training data are used to build the model and test data are used to validate the model. The classification process can be divided into two steps [32] i.e. Learning step and Classification step. In learning step, a classifier is built using an appropriate algorithm and the training data. A classifier is basically a collection of rules produced by the interaction of classification algorithm and training data. The classifier or model obtained from learning step is used in classification step which predicts the class of unknown data. The test data are used here to estimate the accuracy of a model. Categorization is one of the supervised machines learning technique. Machine learning is a self-ruling system which is able to obtaining and combining knowledge continually. This capability of learn from previous experiences, analytical observation, and other means, results in a system that can endlessly self-improve to provide increased efficiency and effectiveness. There are various kinds of machine learning techniques such as Supervised learning, Unsupervised learning, Semi-supervised learning as well as Reinforcement learning. Supervised machine learning techniques can employ what has been learned in the past to new data using labeled examples to predict future events. Beginning from the analysis of an acquainted training dataset, the learning algorithm generates an inferred function to make predictions about the output values. The system is capable of providing targets for any new input after proper training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly [7]. In contrast, unsupervised machine learning algorithms are used when the information used to train is neither classified or unlabeled. Unsupervised learning practice how systems can infer a function to narrate a hidden structure from unlabeled data. The system doesn’t figure out the right output, but it discovers the data and can draw inferences from datasets to narrate hidden structures from unlabeled data. Semi-supervised machine learning techniques fall somewhere in between supervised and unsupervised learning, since they utilize both labeled and unlabeled data for training – generally a small portion of labeled data and a large portion of unlabeled data. The systems that utilize this procedure are capable of considerably improve learning accuracy. Usually, semi-supervised learning is chosen when the obtained labeled data needs skilled and relevant resources in order to train it/learn from it. Otherwise, obtaining unlabeled data typically doesn’t need additional resources. Reinforcement machine learning technique is a learning process that interacts with its environment by producing actions and explores errors or rewards. Trial and error detection and delayed reward are the most relevant features of reinforcement learning. This process permits machines and software agents to automatically identify the ideal characteristics within a specific context in order to maximize its performance. Simple reward feedback is needed for the agent to learn which action is best; this is known as the reinforcement signal [9]. Convolutional Neural Network, also known as CNN or ConvNet, is a class of deep learning neural network which is mostly used in image analysis. It takes image as input, extracts useful features from the image for further implication. It is one of the most powerful images analyzing tools since it requires a minimal preprocessing. CNN uses a variation of multilayer perceptron. The architecture of a ConvNet is similar to the connectivity pattern of Neurons in the Human Brain and was awakened by the formation of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual sector acquainted as the Receptive Field. A collection of such fields overlaps to cover the entire visual area. Convolutional Neural Networks have a different architecture than regular Neural Networks. Regular Neural Networks variate an input by putting it through a series of hidden layers. Every layer is consisting of a set of neurons, where each layer is fully connected to all neurons in the layer before.Finally, the last fully- connected layer known as the output layer represents the predictions.

III. METHODOLOGIES& SYSTEM ARCHITECTURE

Design science research (DRS) can be considered as the lens of synthetic and analysis techniques of Information Science (IS). Generally, it includes two activities to understand the aspect of a system: (1) generation of new knowledge through design and (2) analysis of artifacts [33]. Throughout this paper, DSR method has been pursued and used to enrich the desired model. DSR helps us by providing precise guidelines to evaluate and iterate an artifact within the research project [34].

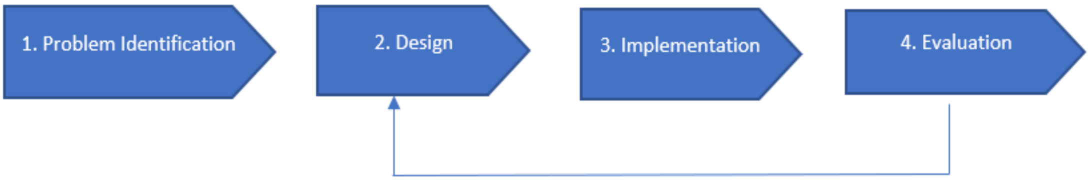

Figure 1: Design Science Research cycle

Functional performance of any system can be improved to get better analysis through the design science research methodology. It can be integrated and employed to different artifacts like algorithms, human-computer interactions etc. We here will utilize the DSR procedure in order to make the model more skilled and useful. The first phase of DSR methodology is problem identification which is shown in figure 1. From where the problems may come is considered in this phase. The formal and informal output of this phase leads to a piloted suggestion. The suggestion then recommends a design for the solution. Then the model goes into the development phase. The artifacts of the system are implemented according to the design and evaluated time to time. Note that, DSR methodology is not a linear procedure. Several phases can be revisited time to time to confirm that the model or system developed is efficient and useful. DSR approach confirms that the artifact is closely linked with the theory and practice.

Categorization is classifying the data for its most useful and feasible use. It is one of the most popular and significant supervised learning techniques in data mining. Let (dj, ci) € D >> C, where D is the collection of documents and C= {c1, c2….c|C|} are set of categories which are reclassified. Then the prominent task of Text Categorization is to assign a Boolean value to each pair in D [13]. Consider the Figure 2, in which D is the Domain of documents and C_1, C_2 and C_3 are various categories. D contains three different kinds of documents such as „@", „#" and „$". After classification, each document is categorized into its respective category.

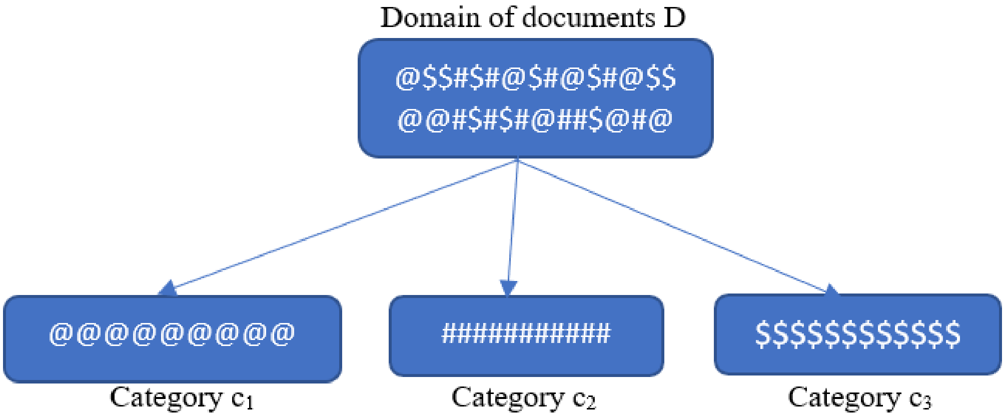

Figure 2: Pictorial representation of Categorization

Text Classification is the task of learning the target function that maps each object set to one of the predefined class labels. Target function is also known as the categorization model. A categorization model aids us to differentiate between objects of various classes [15].A Categorization technique is a chained procedure to construct the categorization model from an input set of data. The technique needs a learning algorithm to pick out a model that understands the relationship between the attribute set and class label of the input data. This learning algorithm should fit the input data well and also predict the class labels of previously unknown records. For constructing any categorization model, a collection of input data set is used. This data set is partitioned into Training Set and Test Set [15].Training data Set means collection of records whose class labels known and is used to construct the classification model. It is then applied to the test dataset. Test data Set means the collection of records whose class labels unknown but when provide an input to the built categorization model, should return the correct class labels of the records. It calculates the accuracy of the model based on the count of correct and incorrect predictions of the test records [15]. There are various categorization techniques in use shown in figure 3.

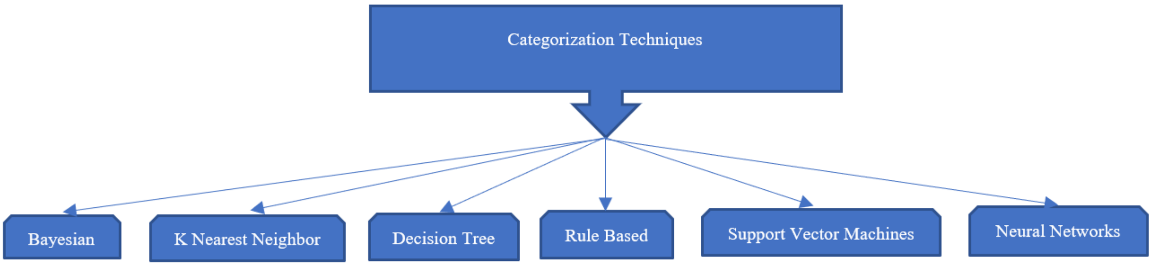

Figure 3: Categorization Techniques.

In this thesis, we will discuss and implement four major categorization techniques, Decision Tree, Support Vector Machine, Bayesian and Nearest Neighbor methodologies. The general procedure for constructing a categorization model is shown in figure 4.

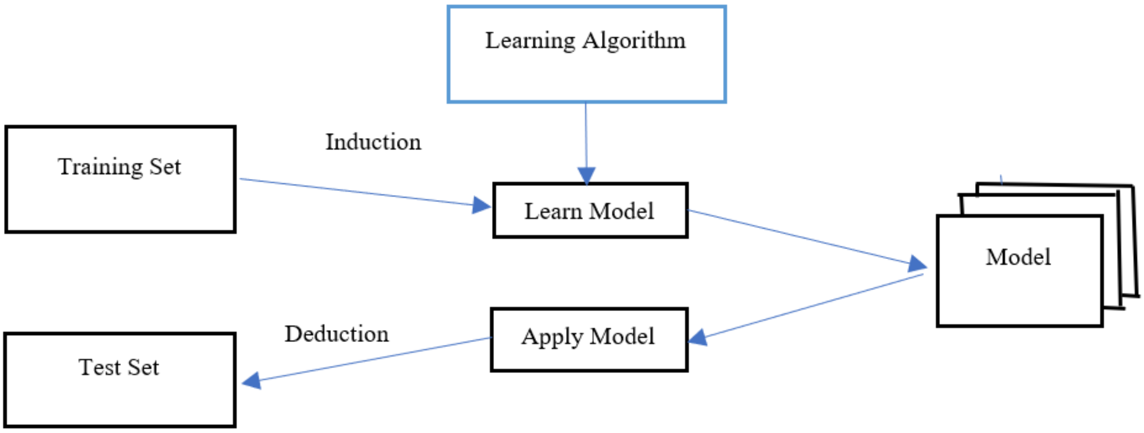

Figure 4: General procedure for constructing a categorization model.

Bayesian is one of the most well-known techniques for classification. It is used to predict the class membership probabilities i.e. probability of a given record belongs to a distinctive category which is based on Bayes Theorem. Bayes theorem is a simple mathematical technique which is used for computing conditional probabilities [14].Let us talk about Bayes Theorem using a small example. X is a sample data record whose category is unknown and H is some assumption. Let sample X belongs to a particular category C. If one needs to determine P(H|X) the probability that the assumption H holds given the data sample X. Bayes Theorem is given as:

$$P(H \mid X) = \frac{P\,(X \mid H)\; P\,(H)}{P(X)}$$

Where P (H|X) is the posterior probability of H on X. Posterior probability is relyed on information such as background knowledge rather than the prior probability which is independent of data sample X. In the same way, P (X|H) is the posterior probability of X on H. If the given data is vast data sample, it would be troublesome to compute above probabilities. Conditional independency was introduced to evacuate this limitation.Naive Bayes categorization is one of the simplest probabilistic Bayesian classification techniques. It is relying on an assumption that the effect of an attribute value on a given class is independent of the values of other attributes which is called as conditional independence. It is used to simplify complex computations [16]. The Naive Bayes classifier is a probabilistic classifier which is based on the Naïve Bayes theorem. From Bayes theorem, the posterior probability can be expressed as:

$P\,(c|x)=\frac{P(c)\ p\ (x|c)}{P(x)}$ cmax yields to the maximum value for P (c|x). The parameter P(c) is calculated as

Where x is a feature vector and x =(x1,…,xn) and c is category. Assume that the category

$$P(c)=\frac{\text{Number of documents in c}}{\text{Number of documents}}$$

The classification results are not influenced because parameter p(x) is independent of categories. Assuming that the elements of feature vectors are statistically independent of each other, p (x|c) can be computed as p(x|c)=Πi p(xi|c), If the maximum estimation is used then

$$P\ (xi|c) = \frac{N\ (xi,c)}{N\ (c)}$$

Where N(x, c) is the joint frequency of x and c,

$$N(c) = \Sigma x N(x,\ c)$$

If some data xi is not presenting in the training data, the probability of any instance containing xi becomes zero, without considering the other features in the vector. Therefore, to ignore zero probability, using Laplacian prior probabilities, p (xi|c) can be computed as follows:

$$p\ (xi|c) = \frac{N\ (xi,\ c) + \lambda}{N(c) + \lambda|V|}$$

Where λ is a positive constant and is chosen as 0.5 or 1, and |V| denotes the number of features.The Naive Bayes categorizer predicts the category cmax with the largest Posterior probability[17]:

$$\begin{aligned} Cmax &= argmaxc\ P\ (c|x) \\ &= argmaxc\ P\ (c)\ p\ (x|c) \end{aligned}$$

NearestNeighbor search is an optimizationproblem for detecting closest points in metric spaces. It is also acquainted as similarity search or closest point search. For a given set of points S in a metric space M and a query point q, the task is to find the closest point in S to q. Typically the distance is determined by Euclidean distance [18].The k-Nearest Neighbor (k-NN) classification is the easiest among all the supervised machine learning techniques but broadly used method for classification and retrieval. It categorizes the objects rely on the closest training examples in the feature space. It is an instance-based learning and often referred as lazy learning algorithm. Here the object instance query is classified rely on the majority of k nearest neighbor category. All the k nearest neighbors in a database of a query are discovered by determining Euclidean distance measure. The neighbors of a query instance are taken from the data set of objects which are already categorized of the classification is previously acquainted [20].The k-Nearest Neighbor categorizer is based on the Euclidean distance between a test sample and the specified training samples. Let X_i be an input sample with P features (x_{i1},x_{i2},……x_{ip}) ,n be the total number of input samples (i = 1,2,…..,n) and P the total number of features (j = 1,2,. ,P). The Euclidean distance between sample x_i and x_l (l = 1,2,….,n) is determined as [21]

$$d(X_i, X_l) = \sqrt{(x_{i1} - x_{l1})^2 + (x_{i2} - x_{l2})^2 + \dots + \left(x_{ip} - x_{lp}\right)^2}.$$

In this way, the class which is illustrated by the largest number of points among the neighbors ought to be the class that the sample belongs to. Nearest Neighbor algorithm is a particular instance of k-NN where k=1 in figure 5. Consider the following figures which represent the sample point in the feature space and neighbors for k = {1, 2, and 3} in figure6 [22].

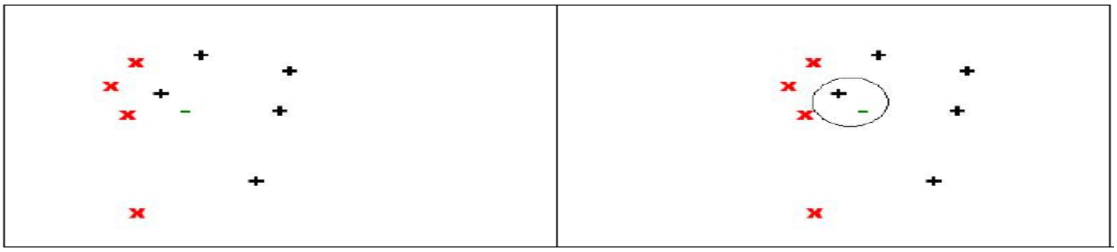

Figure 5: Feature Space when k=1

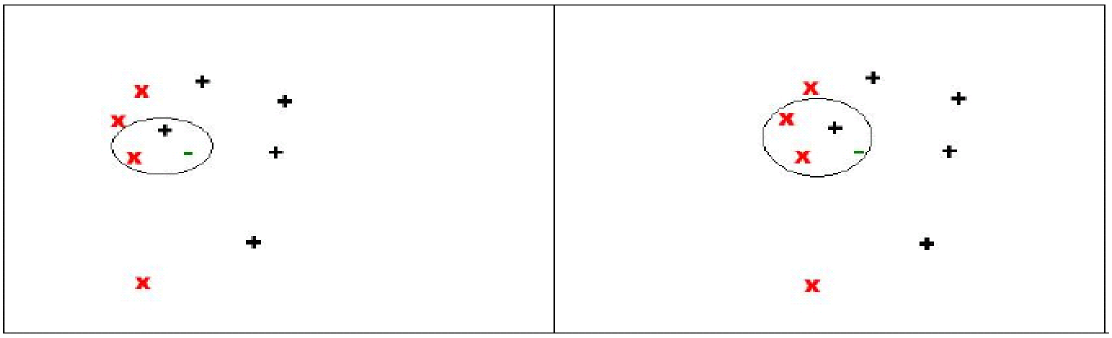

Figure 6: Feature Space when k=2 and k=3

Decision Trees are strong and popular tools for classification and prediction. Decision trees provide rules, which can be understood by humans and used in knowledge system such as database. A decision tree is a hierarchical structure for supervised learning where the local region is professed by a sequence of recursive splits in a smaller number of steps [23]. A decision tree is constructed by internal decision nodes and terminal leaves. Each decision node m implements a test function f(x) with discrete outcomes labelling the branches. Given an input, at each node, a test is employed and one of the branches is selected based on the output. This procedure begins at the root and is repeated recursively until a leaf node is hit, at which point the value written in the leaf constructs the output.A decision tree is also a nonparametric algorithm from the view that we do not assume any parametric form for the class densities and the tree structure is not fixed a priory but the tree grows, branches and leaves are added, when learning depending on the complexity of the problem inherent in the data. Decision tree is a classifier in the shape of a tree structure which consists of [23]:Decision node (individualize a test on a single attribute), Leaf node (tell the value of the target attribute), Edge (divide of one attribute), Path (a disjunction of test to make the extreme decision).Decision trees classifies instances or examples by beginning at the root of the tree and moving through it until a leaf node is met.When decision tree used for text classification it constructs tree where internal node is label by term, branches represent weight and leaf represent the class. Tree can classify the document by going through the query structure from root until it reaches a certain leaf, which express the goal for the classification of the document. Decision tree associated with document is defined as [24] root node which holds all documents, each internal node is subset of documents divided according to one attribute, each arc is labeled with predicate which can be employed to attribute at parent, each leaf node labeled with a class.Decision trees (DT) are the broadly used inductive learning techniques. It is learned from labeled training documents. ID3 is one of the famous decision tree learning algorithms and it has expansions like C4.5 and C5. DT is a flowchart such as tree structures, each internal node reveal test on document, each branch acts outcome of the test, and each leaf node holds a class label.In machine learning, support-vector machine are supervised learning techniques with associated learning algorithms that explore data used for classification and regression analysis. Given a set of training examples, each noted as belonging to one or the other of two categories, an SVM training algorithm constructs a model that assigns new examples to one class or the other, making it a non-probabilistic binary linear classifier. An SVM technique is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as broad as possible. New examples are then mapped into that same space and predicted to belong to a category rely on which side of the gap they fall. From figure 5 we can understand it clearly. In addition to doing linear classification, SVM can also perform non-linear classification efficiently using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

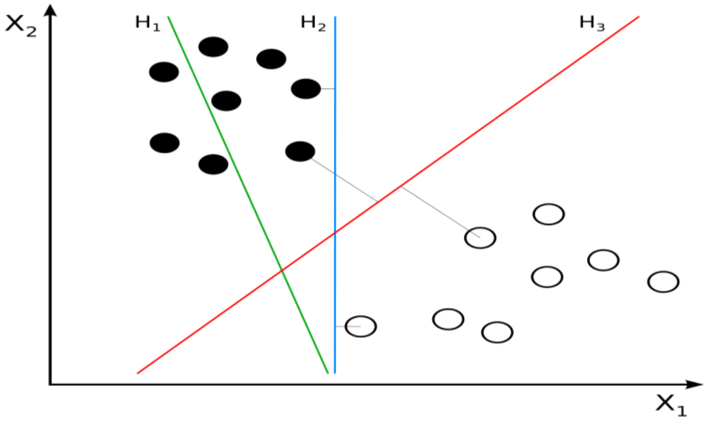

Figure 7: Example of SVM hyperplane pattern.

H_1 cannot separate the classes. H_2 can, but only with a small margin. H_3 divides them with the maximal margin in figure 7.Categorizing data is a general task in machine learning. Suppose some of the given data points belongs to one of two classes, and the goal is to decide which class a new data point will be in. In the case of support-vector machines,a data point is observed as a p-dimensional vector and we want to know whether we can divide such points with a (p-1)-dimensional hyperplane. This is referred as linear classifier. There are many hyperplanes that can classify the data. One reasonable choice as the best hyperplane is the one that express the largest separation, or margin, between the two classes. So, we select the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is referred as the maximum-margin hyperplane and the linear classifier it defines is referred as a maximum-margin classifier; or equivalently, the perceptron of optimal stability.The Support Vector Machine, which was introduced by Vapnik, provides "a maximal margin separating hyper plane" between two classes of data and has non-linear extensions. It is a supervised classification technique which recently utilized successfully for many tasks of NLP as text classification.SVM classifier represents the text document as a vector where the dimension is the number of distinct keywords. If the document size is large then the dimensions are plenty of the hyperspace in text classification which causes high computational cost. The feature extraction and reduction can be used to lessen the dimensionality [25].Our system is divided into two phases containing a training phase and a testing phase. In training phase which is also known as the learning phase, where four classification algorithm builds the classifier by analyzing or learning from a training dataset. In testing phase test data are used to estimate the accuracy of the classification model. If the accuracy is considered acceptable, we can further classify or predicting the class label for new documents by applying our model. Testing phase also known as prediction phase.From the following figure 8 we can easily understand it.

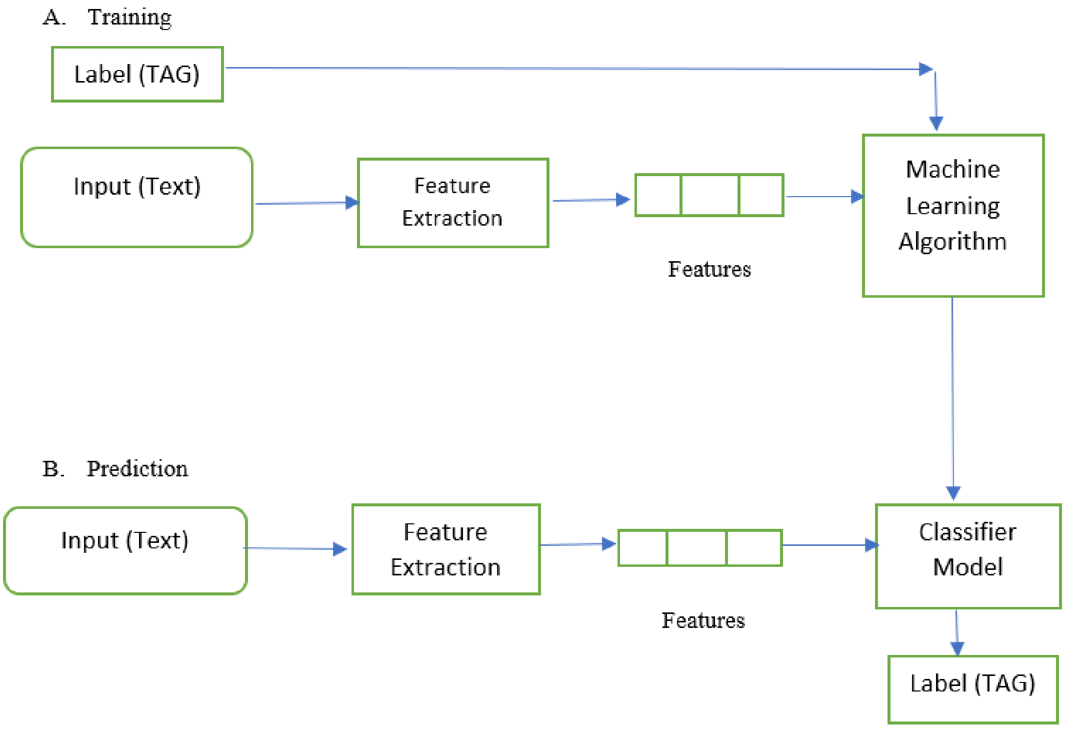

Figure 8: Pictorial representation of System Architecture

IV. SYSTEM IMPLEMENTATION

Thisthesis work mainly focuses on classifying the text documents into their respective categories by employing four major supervised learning techniques such as Naive Bayes, Decision Tree, Support Vector Machine and K- Nearest Neighbor categorizations. We then report on the accuracy and usefulness of the classification by calculating the recall and precision values for each of the categorization models. Python is used as the primary programming language for coding and implementation of these four categorization models.For text classification by using proposed techniques mentioned earlier as Naïve Bayes, Decision Tree, Support Vector Machine and K-Nearest Neighbors classifier, first of all we need a text dataset. For our thesis purpose,we collected “Spam.csv” a text dataset which is available on the internet. It is created by Tiago A. Almeida and José María Gómez Hidalgo [26]. Dataset consists of a collection of 5572 SMS and 2 attributes. The first attribute is class attribute whereas the second attribute is text attribute i.e. SMS. Class attribute has two possible values namely Spam and Ham. Among 5572 SMS, 747 SMS are of type Spam and 4825 SMS are of Ham type. In our thesis work, we mainly concentrated on the second attribute which contains one message per line and choose one category out of two whether the message belongs to ham or spam. Simply Spam SMS is defined as "Unsolicited Bulk Messages". By using Spam SMS, unwanted information is posted to user. This information contains some sort of advertisements, tricks and cheating information. Junk messages are labeled as spam, while legitimate messages are labeled as ham. There is a total of 5572 documents mapped to 2 categories. For learning our classifier, we applied cross validation technique. We used 20% document of total documents as test set and 80% document of total documents as the training set. Training set and Test set collection consisting of 4457 documents and 1115 documents respectively.In this thesis, training documents were used to train the categorization models and the test documents were then employed for the classification of documents into respective categories. Training set with 4457 documents was divided in to 2 categories as spam or ham.As Text Classification is supervised machine learning task since a labelled dataset containing text documents and their labels were used for train a classifier. An end-to-end text classification pipeline is composed of three main elements such as Dataset Preparation, Feature Engineering, Model Training. The first step is the Dataset Preparation step which cover the procedure of loading a dataset and performing basic pre-processing. The dataset is then divided into train and test sets [28].After the dataset has been imported, the next step is to preprocess the text. Text may contain numbers, special characters, stop-words and unwanted spaces. Since, text is the most unstructured form of all the available data, various types of noise are present in it and the data is not readily analyzable without any pre-processing. So, we need to remove these special characters, stop-words and numbers from text. The whole procedure of cleaning and standardization of text, making it noise-free and ready for analysis is known as text preprocessing. Initially pre-processing of the data has been done by using various data mining pre-processing techniques. For pre-processing of data following pre-processing techniques have been used-Transform Case, Tokenize, Removing Stop Words (English), Stemming and Lemmatization.

Tokenize- tokenization is a process which breaks longer strings of text into smaller pieces, or tokens [27]. The further processing is generally performed after a piece of text has been appropriately tokenized.

Transform case – It isused to convert all characters in the text files from upper to lower or to convert from lower to upper case respectively.

Removing Stop words (English)– Stop words like "the", "and", "is", "from" "have" etc. are some of the well-known words which is present in most of the documents [29].It is necessary to remove these stop words because these word does not help to judge the category of the document. This procedure is used to isolate stop words from the document. It is accomplished by deleting every token which is equal to the stop word from the build in stop words list. By removing stop words, space and time can be saved for processing large data.

Stemming - Stemming is the elementary rule-based process of stripping the suffixes ("ing", "ly", "es", "s" etc.) from a word [30]. For example – "write", "writer", "wrote", "writes" and "writing" are the different variations of the word – "write", although they express different but contextually all are same. Stemming transforms all the disparities of a word into their normalized form (also known as lemma).

Lemmatization: Lemmatization, on the other hand, is an embodied & step by step process of obtaining the root form of the word, it makes use of vocabulary (dictionary importance of words) and morphological analysis (word structure and grammar relations).The following figure 9 shows the architecture of text preprocessing pipeline.

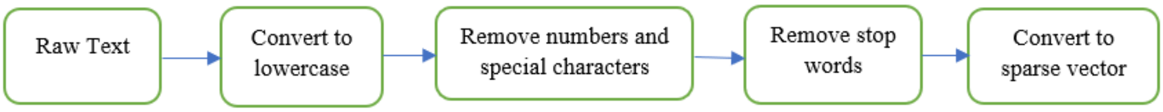

Figure 9: Text cleaning pipeline

The next step is the feature engineering step. In this step, raw text data converted into feature vectors and new features generated using the existing dataset. There is various process in order to acquire relevant features from dataset [28].TF-IDF score represents the relative importance of a term in the document and the entire corpus. TF-IDF score is decided by two terms: the first computes the normalized Term Frequency (TF), the second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears [31].

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).

IDF(t) = $\log_e$ (Total number of documents / Number of documents with term t in it).

TF-IDF Vectors can be generated at different levels of input tokens (words, characters, n-grams).

a) **Word Level TF-IDF:** Matrix illustrate tf-idf scores of every word in different documents.
b) **N-gram Level TF-IDF:** N-grams are the simulation of N terms together. This Matrix represent tf-idf

scores of N-grams.

c) **Character Level TF-IDF:** Matrix representing tf-idf scores of character level n-grams in the corpus[28].

Count Vector is a matrix modulator of the dataset in which every row illustrates a document from the corpus, every column represents a term from the corpus, and every cell illustrates the frequency count of a specific term in a specific document.A word embedding is a form of representing words and documents using a dense vector representation. The position of a term within the vector space is learned from text and is relied on the words that surround the word when it is used. Word embeddings can be trained utilizing the input corpus itself or can be created using pre-trained word embeddings such as Glove, Fast-Text, and Word2Vec.

Besides these, there are many feature engineering techniques which can be used for text classification for identify significant feature. In our thesis for feature engineering we follow count vectorization feature engineering technique [28].

The final step in the text classification framework is to train a classifier using the features created in the previous step. There are many different choices of machine learning models which can be used to train a final model [28]. As we mentioned earlier in our thesis,we will use four supervised machine learning techniques namely Naïve Bayes, Decision Tree, Support Vector Machine and K-Nearest Neighbors technique. We implement these classifiers for this purpose. We can understand the whole process from the following figure 10.

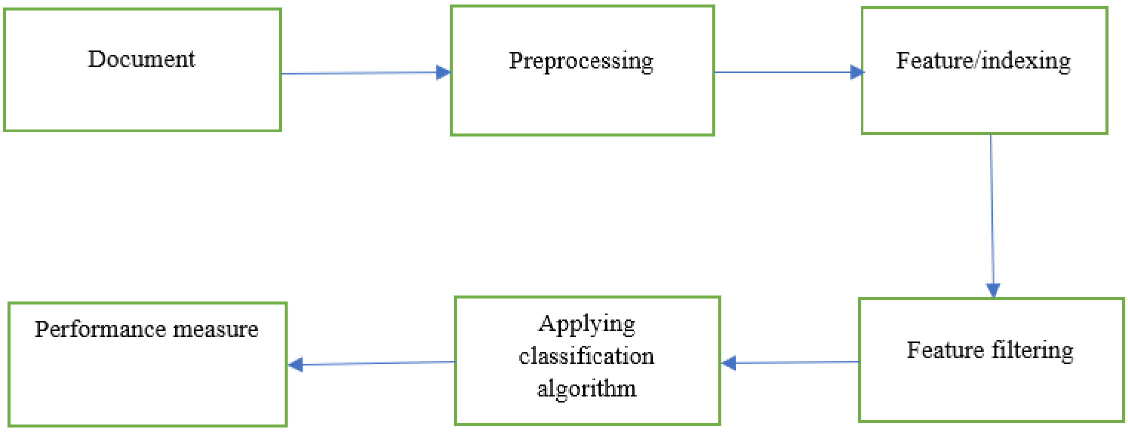

Figure 10: The Text Classification Process.

V. Results and Discussion

In this thesis, wediscussed and illustrated the results obtained from the four algorithms implemented in chapter IV and also evaluated the results to make comparison of these algorithms.Classification evaluation measures include: accuracy, recall, precision and specificity. Before discussing about these, we should know about four building blocks that are used in computing various evaluation measures-True Positives, True Negatives, False Positives, False Negatives.

True Positives (TP): These refers to the positive tuples that were correctly labeled by the classifier. It can be denoted by TP.

True Negatives (TN): These refers to the negative tuples that were correctly labeled by the classifier. It can be denoted by TN.

False Positives (FP): These are the negative tuples that were incorrectly labeled as positive by the classifier. It can be denoted by FP.

False Negatives (FN): These are the positive tuples that were mislabeled as negative by the classifier. It can be denoted by FN.

Accuracy: The accuracy of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier. That is,

$$accuracy = \frac{TP + TN}{P + N}$$

Precision: Precision can be thought of as the measure of exactness (i.e. what percentage of tuples labeled as positive are actually such). It is the fraction of relevant instances among the retrieved instances.

$$precision = \frac{TP}{TP + FP}$$

Recall: Recall is a measure of completeness (what percentage of positive tuples are labeled as such). It is the fraction of relevant instances that have been retrieved over the total amount of relevant instances.

$$recall = \frac{TP}{TP + FN}$$

F Measure: The F measure is a measure of a test's accuracy and is defined as the weighted harmonic mean of the precision and recall of the test.

$$F = \frac{2 * precision * recall}{precision + recall}$$

Confusion Matrix: The confusion matrix is a useful tool for analyzing how well our classifier can recognize tuples of different classes. Each row of the confusion matrix denotes predicted class and each column denotes actual class.

The usefulness of the categorization engines based on Naive Bayes, Decision Tree, Support Vector Machine and K Nearest Neighbor methodologies or the accuracy of the results acquired after the implementation of these algorithms are compared by calculating their standard precision and recall. As we already discussedearlierprecision and recall values evaluates the performance of thecategorization model. There are a total of 1115 test documents where 949 text documents are labeled as ham and 166 text documents is labeled as spam. Let us now have a look at the individual results of each algorithm.

Naïve Bayes Categorization: Out of 1115 test documents given to the Naïve Bayes Categorization model,1100 documents were categorized correctly and the rest of 15 documents were categorized incorrectly. So, the total true positives (TP) for Naïve Bayes are 1100. The confusion matrix which contains true positive, false positive, true negative and false negative for Naïve Bayes classifier is given in table 1.

Actual classPredicted class

	Ham	Spam
Ham (949)	942	7
Spam (166)	8	158

Table 1: Confusion matrix of Naïve Bayes classifier

After gaining all the TP, FP, FN and FP of each category, precision and recall values can be calculated based on the formulae which were discussed earlier. The following table 2 shows the precision and recall of each category for Naïve Bayes classifier.

	Precision	Recall
Ham	0.99	0.99
Spam	0.96	0.95

Table 2: Precision and Recall values of Naïve Bayes classifier

The standard Precision and Recall values obtained for Naïve Bayes categorization are 0.99 and 0.98 respectively. This implies that, based on Naive Bayes methodology our categorization model shows 99% exactness and 98% completeness of accuracy levels.

Support Vector Machine Categorization: Out of 1115 test documents given to the Support Vector Machine model,1080 documents were categorized correctly and the rest of 35 documents were categorized incorrectly. So, the total true positives (TP) for Support Vector Machine model are 1080.The confusion matrix which contains true positive, false positive, true negative and false negative for SVM classifier is given in table 3.

Actual class Predicted class

	Ham	Spam
Ham (949)	943	6
Spam (166)	29	137

Table 3: Confusion matrix of SVM classifier

After gaining all the TP, FP, FN and FN of each category, precision and recall values can be calculated based on the formulae which were discussed earlier. The following table 4 shows the precision and recall of each category for SVM classifier.

	Precision	Recall
Ham	0.98	1.00
Spam	0.99	0.87

Table 4: Precision and Recall values of SVM classifier

The standard Precision and Recall values obtained for SVM categorization are 0.98 and 0.98 respectively. This implies that, based on Support Vector Machine methodology our categorization model shows 98% exactness and 98% completeness of accuracy levels.

k -Nearest Neighbor Categorization: Out of 1115 test documents given to the k-Nearest Neighbor Categorization model,1025 documents were categorized correctly and the rest of 90 documents were categorized incorrectly. So, the total true positives (TP) for k Nearest Neighbor Categorization model are 1025.The confusion matrix which contains true positive, false positive, true negative and false negative for k-Nearest Neighbor Categorization classifier is given in table 5.

Actual class Predicted class

	Ham	Spam
Ham (949)	949	0
Spam (166)	90	76

Table 5: Confusion matrix of K-NN classifier

After gaining all the TP, FP, FN and FN of each category, precision and recall values can be calculated based on the formulae which were discussed earlier. The following table 6 shows the precision and recall of each category for K-NN classifier.

	Precision	Recall
Ham	0.91	1.00
Spam	1.00	0.46

Table 6: Precision and Recall values of K-NN classifier

The standard Precision and Recall values obtained for K-NN categorization are 0.92 and 0.92 respectively. This implies that, based on K-NN methodology our categorization model shows 92% exactness and 92% completeness of accuracy levels.

Decision Tree Categorization: Out of 1115 test documents given to the Decision Tree Categorization model,1080 documents were categorized correctly and the rest of 35 documents were categorized incorrectly. So, the total true positives (TP) for Decision Tree Categorization model are 1080.The confusion matrix which contains true positive, false positive, true negative and false negative for Decision Tree Categorization classifier is given in table 7.

Actual class Predicted class

	Ham	Spam
Ham (949)	943	6
Spam (166)	29	137

Table 7: Confusion matrix of Decision Tree Categorization classifier

After gaining all the TP, FP,TN and FN of each category, precision and recall values can be calculated based on the formulae which were discussed earlier. The following table 8 shows the precision and recall of each category for Decision Tree Categorization classifier.

	Precision	Recall
Ham	0.97	0.99
Spam	0.96	0.83

Table 8: Precision and Recall values of Decision Tree Categorization classifier

The standard Precision and Recall values obtained for Decision Tree Categorization are 0.97 and 0.97 respectively. This implies that, based on Decision Tree methodology our categorization model shows 97% exactness and 97% completeness of accuracy levels.

After getting the standard precision and recall for each classifier, we can compute the accuracy of each classifier and can find out the best classifier by comparing the accuracy of each classifier. The accuracy obtained for four classifiers are shown in the following table 9:

Classifier Name	Accuracy (%)
Naïve Bayes	98.66
Support Vector Machine	98.02
Decision Tree	96.86
K-Nearest Neighbor	91.93

Table 9: Accuracy of four classifiers

From the accuracy table of four classifiers, we see that Naïve Bayes and SVM techniques obtained best accuracy than Decision Tree and K-NN classifiers. After comparing four classifiers, we see that Naïve Bayes and SVM approach for text classification with processed dataset leads to higher accuracy because of organized preprocessing.

VI. CONCLUSION AND FUTURE WORK

In this thesis, "Comparative Study on Text Classification" we studied the four basic classification techniques and developed four categorization models based on Support Vector Machine, Naïve Bayes, Decision Tree and k -Nearest Neighbor methodologies.In chapter II, we discussed the background of the classification where various machine learning approach are defined and narrated theoretically. Moreover, we also discussed various research work which have been done and going on in our selected topic. In chapter III, we discussed the methodologies& system architecture of our research work. We showed general approach of text categorization and try to discussed about our proposed supervised learning algorithms.In chapter IV, we discussed about our whole implementation process to build the classifier. For our thesis purpose "Spam.csv" text dataset is collected which is available on the internet. Various kinds of preprocessing procedure employed on the documents and also for implementing four classifiers are described in chapter IV. In chapter V, our obtained experimental results are tabulated, compared and evaluated. We compared the usefulness of Naive Bayes, Support Vector Machine, Decision Tree and K-NN categorization engines by evaluating their standard precision, recall and based on their individual model accuracy. From our whole study, we noticed that the standard precision and recall values of Naïve Bayes and Support Vector Machine categorization engine are better than Decision Tree and K-NN engines.Text classification is a contemplative area of research in the field of information retrieval and machine learning. In our future work, we willimplement more classification models and compare them with existing models and also determine which has the best accuracy.

REFERENCES

[1] M Ozaki, Y. Adachi, Y. Iwahori, and N. Ishii, Application of fuzzy theory to writer recognition of Chinese characters, *International Journal of Modelling and Simulation, 18(2),* 1998, 112-116.

[2] Saleem, H."Information Retrieval: A framework for Recommending text based classification algorithms". Pace University, Ph.D.Thesis, 2002.

[3] Witten, I.H., Frank, E. and Hall, M.A. *Data Mining Practical Machine Learning Tools and Techniques*, ,2011.

[4] Olson, D.L. and Delen, D. *Advanced Data Mining Techniques.* Springer-Verlag Berlin Heidelberg, German, 2008.

[5] Khan, A., Baharudin B., Lee, L.H. and A.Khan, K.K. "A Review of Machine Learning Algorithms for Text- Document.

[6] B. Tang, S. Kay and H. He, "Toward Optimal Feature Selection in Naive Bayes for Text Categorization," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 9, pp. 2508-2521, 1 Sept. 2016.

[7] Wikipedia, the free Encyclopedia, Supervised Machine Learning. http://en.wikipedia.org/wiki/Supervised_learning

[8] Bo Tang, Haibo He, Paul M. Baggenstoss, Steven Kay, "A Bayesian Classification Approach Using Class-Specific Features for Text Categorization", *Knowledge and Data Engineering IEEE Transactions on*, vol. 28, no. 6, pp. 1602-1606, 2016.

[9] Wikipedia, the free Encyclopedia, Machine Learning. http://en.wikipedia.org/wiki/Machine_learning

[10] Ardhapure, Omkar, Linganagouda S. Patil, Disha Udani and Kamlesh Jetha. "COMPARATIVE STUDY OF CLASSIFICATION ALGORITHM FOR TEXT BASED CATEGORIZATION." IJRET:Feb(2016).

[11] Karamcheti, Aditya Chainulu, "A Comparative study on text categorization" (2010). *UNLV Theses, Dissertations, Professional Papers, and Capstones*. 322. https://digitalscholarship.unlv.edu/thesesdissertations/322.

[12] AadityaJain,JyotiMandowara, "Text classification by combining text classifiers to improve the efficiency of classification". IJCA: April(2016).

[13] Fabrizio Sebastiani, 'Machine Learning in Automated Text Categorization', Italy2002. http://nmis.isti.cnr.it/sebastiani/Publications/ACMCS02.pdf.

[14] James Joyce, 'Bayes Theorem' Stanford Encyclopedia of Philosophy, June 2003. http://plato.stanford.edu/entries/bayes-theorem

[15] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, 'Introduction to Data mining', Chapter 4 Pearson Addison Wesley, 2005.

[16] Wikipedia, the free Encyclopedia, Naive Bayes Classifier, 2008. http://en.wikipedia.org/wiki/Naive_Bayesian_classification

[17] Thesis on „Clustering Approaches to Text Categorization'' by HiroyaTakamura
http://www.lr.pi.titech.ac.jp/~takamura/pubs/dthesis_original.pdf

[18] Wikipedia, the free Encyclopedia, Nearest neighbor search.
http://en.wikipedia.org/wiki/Nearest_neighbor_search#Knearest_neighbor

[19] Text Categorization with SVM: Learning with Many Relevant Features by Thorsten Joachims. http://www.cs.cornell.edu/people/tj/publications/joachims_98a.pdf

[20] Wikipedia, the free Encylcopedia, K Nerest neighbor Algorithm. http://en.wikipedia.org/wiki/K-nearest_neighbor_algorithm

[21] Scholarpedia, K Nearest neighbor. http://www.scholarpedia.org/article/K-nearest_neighbor

[22] k-Nearest Neighbor (kNN) Algorithm. https://kiwi.ecn.purdue.edu/rhea/index.php/KNN_Algorithm_Old_Kiwi

[23] Wikipedia, the free Encyclopedia, Decision Tree. https://en.wikipedia.org/wiki/Decision_tree

[24] Gandhi, V. C., & Prajapati, J. A. (2012). Review on Comparison between Text Classification Algorithms. International Journal.

[25] Wikipedia, the free Encyclopedia, Support Vector Machine. https://en.wikipedia.org/wiki/Support-vector_machine

[26] http://www.cs.tau.ac.il/~arielhe/ml12-project/

[27] https://www.kdnuggets.com/2017/12/general-approach-preprocessing-text -data.html

[28] https://www.analyticsvidhya.com/blog/2018/04/a-comprehensive-guide-to-understand-and-implement-text-classification-in-python/

[29] https://en.m.wikipedia.org/wiki/Stop_words

[30] Porter Stemmer Algorithm. http://tartarus.org/~martin/PorterStemmer/

[31] Term Frequency and Inverse Document Frequency – Wikipedia. http://en.wikipedia.org/wiki/Term_frequency

[32] Jiawei Han, Micheline Kamber, Jian, Pei, "Data Mining Concepts and Techniques", Third Edition, Morgan Kaufmann Publisher, 2012.

[33] Vijay Vaishnavi, Bill Kuechler, Stacie Petter, "DESIGN SCIENCE RESEARCH IN INFORMATION SYSTEMS", IEEE Transactions on Knowledge and Data Engineering (TKDE).

[34] Gacenga, F., Cater-Steel, A., Toleman, M. and Tan, W.G., 2012. A proposal and evaluation of a design method in design science research. Electronic Journal of Business Research Methods, 10(2), pp.89-100.

Amina Khatun, et. al. "Comparative Study on Text Classification." *International Journal of Engineering Science Invention (IJESI)*, Vol. 09(09), 2020, PP 21-33. Journal DOI- 10.35629/6734