International Journal Of Engineering Research And Development
e- ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 16, Issue 4 (April 2020), PP. 48-55

Document Classification using Machine Learning Techniques

Rajnish Virpate[1], Adityavikram Gurao[2], Ankit Naik[3], Mahebub Shaikh[4]
Smita Chaudhari[5]

UG Student[1], Department of Computer Engineering, DIT, Pimpri, Pune, India
UG Student[2], Department of Computer Engineering, DIT, Pimpri, Pune, India
UG Student[3], Department of Computer Engineering, DIT, Pimpri, Pune, India
UG Student[4], Department of Computer Engineering, DIT, Pimpri, Pune, India
Assist.Prof[5], Department of Computer Engineering, DIT, Pimpri, Pune, India

Abstract:-*Automated classification of text documents into their meaningful classes has always been considered as a vital method to manage and process a vast amount of documents in digital forms that are widespread and continuously increasing. This kind of web information, popularly known as the digital/electronic information is in the form of documents, conference material, publications, journals, editorials, web pages, e-mail etc. People largely access information from these online sources rather than being limited to archaic paper sources like books, magazines, newspapers etc. But the main problem is that this enormous information lacks organized nature which makes it difficult to manage. Document classification is recognized as one of the key techniques used for organizing such kind of digital data. Document classification is an example of Machine Learning (ML) in the form of Natural Language Processing (NLP). By classifying text, we are aiming to assign one or more classes or categories to a document, making it easier to manage and sort. This is especially useful for publishers, news sites, blogs or anyone who deals with a lot of content. Here, we propose a document classification system that can classify documents into their meaningful classes in which documents are very likely to have similar subjects. The proposed system extracts data by pre-processingand extracts super-topics and subtopics with the help oftf-idfand Pachinko Allocation Model (PAM) scheme. Then, the Naive Bayes Classifier is applied to classify, whole documents into documents with similar subjects.*

Keywords: -*NLP , PAM , TF-IDF , Naive Bayes.*

Date of Submission: 28-05-2020 Date of Acceptance: 14-06-2020

I. INTRODUCTION

Document classification is that the task of grouping documents into categories based upon their content. Document classification could be a significant learning problem that's at the core of the many information management and retrieval tasks.

Document classification performs an important role in various applications that deals with organizing, classifying, searching and concisely representing a big amount of data. Document classification could be a longstanding problem in information retrieval which has been well studied. Therefore, it's desired that these huge numbers of documents are systematically classified with similar subjects in order that users can find interested documents easily and conveniently. Typically, finding documents on specific topics or subjects is time consuming activity. The commonly-used analysis for the classification of an enormous number of documents is run on large-scale computing machines with none consideration on big data properties.

As time goes on, it's difficult to manage and process efficiently those documents that still quantitatively increase. Since the relation of the documents to be analyzed and classified is extremely complex, it's also difficult to catch quickly the topic of every document and, moreover hard to accurately classify document with the similar subjects in terms of contents. Therefore, there's a requirement to use an automatic processing method for such an enormous number of documents in order that they're classified fast and accurately.

The rest of the paper is organized as follows. Section 2 focuses on the Literature Survey. Section 3 presents our proposed approach for document classification. Section 4 focuses on Experimental results. Section 5 presents the Conclusion and Future Work. Section 6 describes the References.

II. RELATED WORK

In this section, we primarily aim to investigate empirical methods that are used for the classification of documents.

Sang-woon Kim and Joon-Min Gil proposed classification system based on TF-IDF and LDA schemes[1]. Consequently an abundance of approaches has been developed for such purposes, including classification system based on TF-IDF and LDA schemes. This paper has presented a technique for automatic text classification, which incorporates phases like pre-processing, topic modelling (LDA scheme), feature selection and selecting K-Mean clustering as machine learning technique for the classification. they need also discussed a number of the key issues involved in text classification like handling great amount of information, handling classification of documents supported the subjects obtained from the topic model.

Andrew McCallum and Wei Li has presented Pachinko Allocation Model(PAM) an improved version of the topic model Latent Dirichlet Allocation (LDA)[2]. LDA represents each document as a mixture of topics, where each topic is a multinomial distribution over words in a vocabulary. to get a document, LDA first samples a per-document multinomial distribution over topics from a Dirichlet distribution. Then it repeatedly samples a topic from this multinomial and samples a word from the topic. PAM provides more flexibility and greater expressive power than latent Dirichlet allocation.

Md.Saiful Islam, FazlaElahiMdJubayer and Syed Ikhtiar Ahmed has presented the research which uses SVM machine learning technique to categorize the documents with the TFIDF algorithm[3]. in this research paper they have described that SVM with TFIDF and word unigram performs well compared to the bigram, trigram and char-ngram . because the dataset increases, the accuracy increases.

AnkitaDhar, NiladriSekhar Dash and Kaushik Roy have presented TF-IDF-ICF text analysis scheme which is an improved version of TF-IDF scheme[4]. This paper shows that the application of TF-IDF-ICF feature with dimensionality reduction technique can bring in precision in classifying the text documents to their respective categories. From this experiment, it's evident that based on this reduction technique adopted for Bangla text documents classification, it's possible to attain high accuracy. In future, the system are often tested on a bigger dataset with an oversized number of text categories. Also, other commonly used standard reduction techniques are often applied along side different feature extraction and selection techniques also.

Mowafy M, Rezk A, El-bakry HM proposed the research where they stated that consistent with the chosen embedded techniques, it can improve the performance of the opposite techniques[5]. because it is predicated on an efficient model for Unstructured Text Document in which they have presented a model for text classification that depicts the stream of phases through building automatic text document classifier and presenting the connection between them. They have stated that consistent with the chosen embedded techniques it can improve the performance of other techniques, since the accuracy of KNN –TFIDF can be improved.

PuneetGoswami and Vidya Kamath have proposed the research based on TF-IDF algorithm – modified TF-IDF[6]. The tf-idf is an algorithm which is usually used where massive processing is completed. Tf-idf is the weight given to a specific term within a document and it's proportional to the importance of the term. in this paper an attempt has been made to propose an algorithm which might be used to find the importance of the document. the idea here is that the importance of a document increases with the amount of times it's viewed similar to the way the importance of a term increases with its number of occurrences. The tf-idf algorithm has been used and an idea has been extended to design the df-icf algorithm. Calculating tf-idf is extremely important since it's utilized in situations where massive data processing is completed. therefore the tf-idf needs to be done quicker and faster. If we discover the df-icf first, you'll be able to filter the documents which are of less importance and avoid the calculations of tf-idf within them. This not only saves some time but also makes your result more reliable. If we will efficiently apply the df-icf algorithm to large data sets then you'll save your effort, time and resources.

Ms.SharmilaShinde, Dr.PrasannaJoeg and Dr. Sandeep Vanjale have presented the research based on web document classification using support vector machine[7]. In their paper, they have proposed a web page classification method using an SVM. The experimental results show that the linear kernel function yields the simplest results of these four kernel functions. When using the training documents as a test set, the results of SVM are giving good classification accuracy. The experiment also demonstrated that SVM yields better accuracy once we tuned kernel function with 0.5 gamma value.

KanikaDhanjal and SangitaBhagat proposed a way for news articles categorization using SVM and KNN with TF-IDF approach[8]. It's a supervised learning approach during which news articles are assigned category labels supported likelihood demonstrated by a training set of labeled articles. They used TF-IDF term weighting scheme for optimization of classification techniques to induce more optimized results and use two supervised learning approaches, i.e., SVM and KNN and compare the performances of both classifiers.

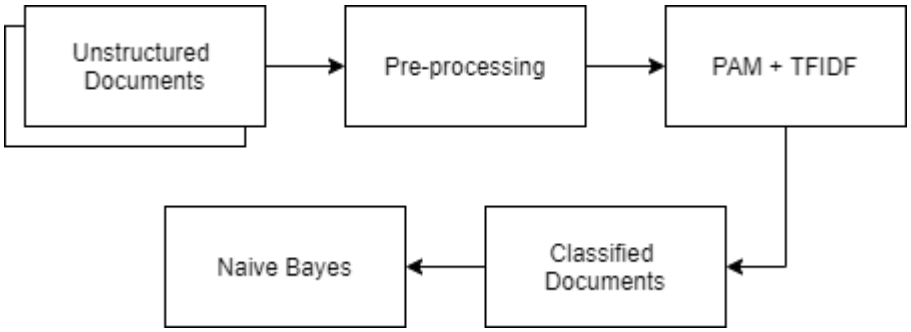
Manjula S. and Ravindra S. Hegadi have proposed Identification and Classification of Multilingual Document using Maximized Mutual Information[9]. Their paper is addressing the matter of detecting document that consists of quite one language by using maximized mutual information technique, identified languages are

classified by implementing KNN and SVM classification model. Indian languages have their own characteristics and that they are often distinguished with the assistance of visual discrimination methods and statistical methods. to spot these differences through machine they're making use of edge direction based feature. To capture the differences present in languages Edge Direction Histogram (EDH) is employed. they need proposed this system for identification and classification of Indian languages present in multilingual document by using maximized mutual information and classification is completed supported different classification model. Document classification or document categorization is a drag in library science, information science and computer science. The task is to assign a document to one or more classes or categories. this might be done "manually" (or "intellectually") or algorithmically.

Mingyong Liu and Jiangang Yang have presented the improvement of TFIDF weighting in text categorization[10]. They have described the problem of to improve the classification accuracy in text categorization. they have stated that so as to unravel the matter of accuracy of classification they tried to enhance the accuracy by proposing an improvement on TF-IDF weighting method.

III. PROPOSED APPROACH

It is essential to perform tokenization which is basically the segmentation of sentences into tokens and pre-processing where the tokens that carry no relevant domain-specific information are removed . The following block diagram (Fig. 1) demonstrates the overall process of automatic document classification system implemented in this experiment. A detailed description of the model is provided below.



1. At this step unstructured files are taken as input for the document classification process.
2. At the pre-processing stage, elimination of stop words, punctuations, postpositions, conjunctions, and other processes are performed. Data pre-processing includes cleaning, Instance selection, normalization, transformation, feature extraction, etc. Tokenization is the process of converting a sequence of characters (such as in a computer program or web page) into a sequence of tokens (strings with an assigned and thus identified meaning).Stop words are words which are filtered out before or after processing of natural language data (text).
3. This is the processing stage where tf-idf(feature selection scheme) and PAM(topic model scheme) are used. a subject model may be a sort of statistical model for locating the abstract "topics" that occur during a collection of documents. Pachinko Allocation Model(PAM) is employed to uncover the hidden thematic structure of a set of documents whereas TF-IDF helps in increasing the accuracy of the result.
4. At the classification process during which ideas and objects are recognized, differentiated and understood. For the classification Naive Bayes method is employed which helps in classifying documents based upon the probability produced because the output by step 3.
5. This is the ultimate step where we obtain the classified document as per the need.

Data pre-processing is implemented in step 2 during which data is pre-processed using various methods. Data-gathering methods are often loosely controlled, leading to out-of-range value, impossible data combinations, missing values, etc. Analyzing data that has not been carefully screened for such problems can produce misleading results. Thus, the representation and quality of knowledge is first and foremost before running an analysis. Often, data pre-processing is the most vital phase of a machine learning project, especially in computational biology.

If there's much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery during the training phase is harder. Data preparation and filtering steps can take a considerable amount of time interval.The product of knowledge pre-processing is that the final training set.Data pre-processing may affect the way during which outcomes of the ultimate processing are often interpreted.

In step 3, TF-IDF has been wont to increase the efficiency of PAM. TF-IDF is widely utilized in the fields of data retrieval and text mining to gauge the connection for every word within the collection of documents. especially, they're used for extracting core words from documents.

The TF(Term frequency) in TF-IDF means the occurrence of specific words in documents. Words with a high TF value have an importance in documents.

$$TF_{i,j} = \frac{n_{i,j}}{\sum_{k} n_{k,j}}$$

On the other hand, the DF implies what percentage of a selected word appears within the collection of documents. It calculates the occurrence of the word in multiple documents, not in just a document. Words with a high DF value don't have an importance because they commonly appear in documents.

$$DF_{i,j} = \frac{d_j \in D : t_j \in d_j}{|D|}$$

Accordingly, the IDF that is an inverse of the DF is used to measure the importance of words in all documents. The high IDF values mean rare words in all documents, resulting to the increase of an importance. TF-IDF is basically used to increase the accuracy of the result which will be produced with the help of PAM(topic model).

In Step 3 PAM is implemented. LDA imagines a hard and fast set of topics. Each topic represents a group of words. and therefore the goal of LDA is to map all the documents to the topics during a way, such the words in each document are mostly captured by those imaginary topics. But it still has some shortcomings and to satisfy those shortcomings and improve the accuracy of the document classifier we use the PAM model.

Pachinko Allocation Model (PAM), it's a topic modeling technique which is an improvement over the shortcomings of Latent DirichletAllocation.Inability to extract the connection between topics poses an enormous limitation of LDA method. Because during a continuous passage, the succeeding line would have certain coherence with its preceding line. So it's highly important to get tight coherence between passages to get proper topics. This difficulty is overcome by using Pachinko Allocation Model.

It captures arbitrary, nested and even sparse correlation between topics using Directed Acyclic Graph. The list of all words obtained from the corpus after removing the stop-words and text processing represents the Dirichlet distribution. In a PAM, each of the subject generated is related to the Dirichlet distribution through a Directed Acyclic Graph (DAG). In this model, each leaf node corresponds to the words present within the vocabulary and every non-leaf node represents the topics. In an arbitrary DAG, a LDA model wouldn't have the links between those non leaf interior nodes where the subsequent image explains the structural arrangement of the PAM model.

In the final step we make use of Naive Bayes for classification. Naive Bayes classification makes use of Bayes theorem to work out how probable it's that an item may be a member of a category. Naive Bayes sorts items into categories supported whichever probability is highest. Bayes theorem tells us that the probability of a hypothesis given some evidence is adequate to the probability of the hypothesis multiplied by the probability of the evidence given the hypothesis, then divided by the probability of the evidence.

$$\Pr(H|E) = \Pr(H) * \Pr(E|H) / \Pr(E)$$

It's "naive" because it treats the probability of every word appearing during a document as if it were independent of the probability of the other word appearing. Once we follow these rules, some words tend to be correlated with other words.

IV. EXPERIMENTAL RESULTS

For this experiment, we are using unstructured data files as input. These input files are initially pre-processed to get rid of irrelevant and redundant information present or noisy and unreliable data to decrease the problem of data gathering during the training phase.In pre-processing, tokenization and stop word removal are implemented to get rid of irrelevant information.

Tokenization and Stop word removal are implemented with the assistance of nltk(natural language toolkit) library.

$$stop\_words = set(stopwords.words('english'))$$
$$tokens = word\_tokenize(line)$$

After being pre-processed the info is processed through TF-IDF + PAM structure. The TF-IDF weighting scheme consists of two terms Term Frequency(TF) and IDF(Inverse Document Frequency). TF here represents the frequency of a term t during a particular text document d. Whereas IDF generally measures the relevance of a term t in d. Here we are using TF-IDF to extend the accuracy of the output produced by the Pachinko Allocation Model (PAM) structure.

```
def PAM_EXAMPLE(input_file,save_path):                                    (1)
mdl = tp.PAModel(tw = tp. Term Weight.ONE, min_of = 3, rm_top = 5, kl = 1, k2 = 5, alpha = 0.1, eta = 0.01)
for n, line in enumerate(open(input_file, encoding ='utf -8')):
ch = line.strip().split()
mdl.add_doc(ch)
mdl.burn_in = 100
mdl.train(0)
print('NUM DOCS:', len(mdl.docs),', vocabsize:', mdl.num_vocabs,', Num words :', mdl.num_words)
print('Removed top words:', mdl.removed_top_words)
print('Training...', file = sys.stderr, flush = True)
for i in range(0,1000,10):                                                (2)
mdl.train(10)
print('Iteration: {} \ tLog - likelihood: {}'.format(i, mdl.ll_per_word))
print('Saving...', file = sys.stderr, flush = True)
mdl.save(save_path,True)
for k1 in range(mdl.k1):                                                  (3)
print('Super_Topic # {}'.format(k1))
for sub in mdl.get_sub_topics(k1):
print('\t', sub)
for k2 in range(mdl.k2):                                                  (4)
print('Sub_topic_words#{}'.format(k2))
or word, prob in mdl.get_topic_words(k2, top_n =10):
print('\t', word, prob, sep ='\t')
```

It utilizes a vectorization of recent CPUs for maximizing performance. We make use of avx2 instruction set to get faster training iterations of log likelihood which ends up in faster performance. So, basically it exploits AVX2,AVXor SSE2 SMID instruction set which may end in faster iterations. It takes advantage of multicore CPUs with a SIMD instruction set, which may end in faster iterations.

We used Collapsed Gibbs-Sampling(CGS) to infer the distribution of topics and therefore the distribution of words to calculate the likelihood of heldout data, we must integrate out the sampled multinomials and sum over all possible topic assignments.This problem has no closed-form solution. Previous work that uses Gibbs sampling for inference approximates the likelihood of a document d by the mean of a group of conditional probabilities, where the samples are generated using Gibbs sampling.

$$P\left(d\middle|z\left(d\right)\right)$$

After being processed through the PAM structure super-topics(Table 1) and sub-topic words(Table 2) are generated with their own probabilities which are later processed through naive bayes classifier which helps in classifying the documents.

Super-topic #0	
Sub-topic #1	0.214
Sub-topic #2	0.203
Sub-topic #0	0.202
Sub-topic #3	0.201
Sub-topic #4	0.178

Table.1: Super-topics

Sub-Topic-Words #0	Probability	Sub-Topic-Words #1	Probability	Sub-Topic-Words #2	Probability	Sub-Topic-Words #3	Probability	Sub-Topic-Words #4	Probability
one	0.010	produce	0.007	new	0.007	work	0.010	time	0.007
found	0.007	city	0.006	many	0.006	would	0.009	number	0.006
year	0.006	follow	0.005	gener	0.005	day	0.007	well	0.005
made	0.006	french	0.005	often	0.005	system	0.007	power	0.005
may	0.006	since	0.005	centuri	0.005	part	0.006	history	0.005
de	0.005	server	0.005	year	0.005	people	0.006	school	0.005
differ	0.005	product	0.005	include	0.005	include	0.005	th	0.004
two	0.005	make	0.005	area	0.005	known	0.005	may	0.004
th	0.005	play	0.004	form	0.005	like	0.005	intern	0.004
base	0.005	end	0.004	call	0.005	name	0.005	nature	0.004

Table.2: Sub-topics

Supertopic are often described as a subject that encompasses several other topics, whereas subtopic may be a subject that forms a part of a subject.After being processed through the PAM structure multiple super topics are obtained. Super topic encompasses other topics which are referred to as sub topics.Sub topic is essentially a neighborhood of super topic.

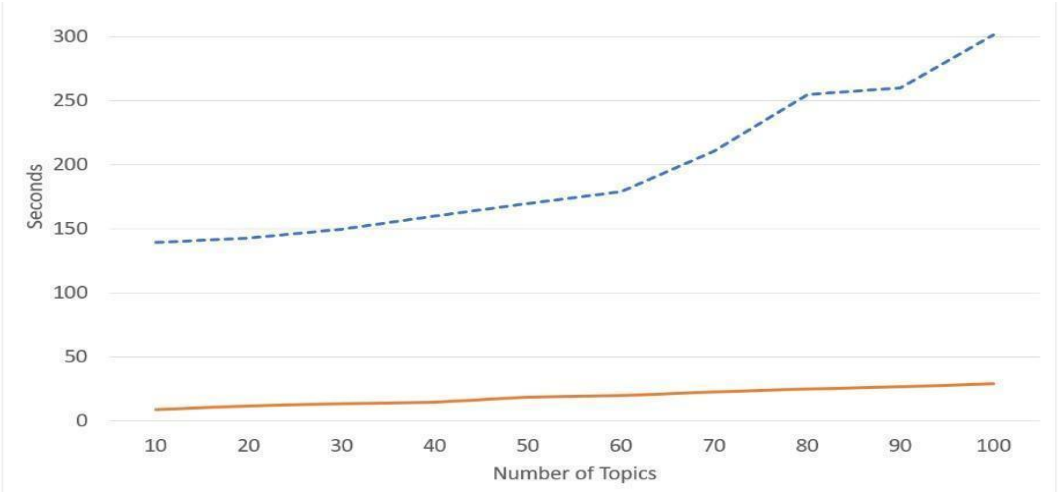


Figure (1)

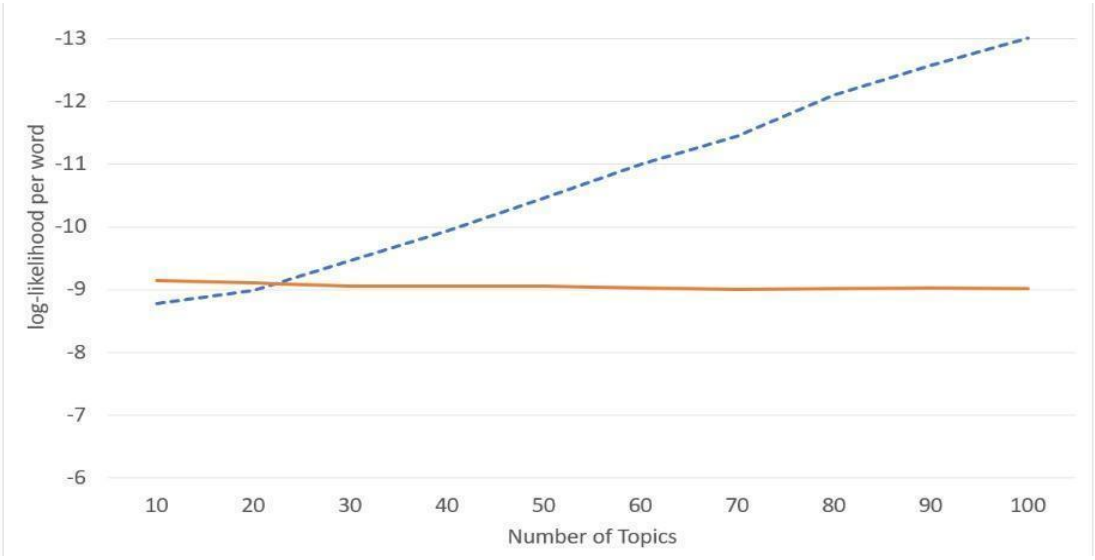


Figure (2)

In the above graphs we compare the proposed model and regular model which helps us in acquiring the info which tells us about the efficiency of the proposed model compared to regular model.We use log-likelihood to match the proposed model and a daily model.

The log-likelihood function is typically used to derive the maximum likelihood estimator of the parameter. The negative log-likelihood becomes unhappy at smaller values, where it can reach infinite unhappiness, and becomes less unhappy at larger values, the higher value is better.For example, a log-likelihood value of -3 is best than -7.

Fig(1) states , number of topics obtained during a specific amount of time(seconds) where line is our proposed model whereas dashed line is regular model. Fig(2) states, log-likelihood values for every iteration . line is our proposed model whereas dashed line is regular model. Negative value keeps increasing for normal model which makes it more unhappy compared to our proposed model which has higher value. Overall the above graphs provides us the info which tells us our proposed model is more efficient compared to regular models.

V. CONCLUSION AND FUTURE WORK

Document Classification deals with the representation, organization of document and access to information items in an ordered way. In mass recruitment, this software are often wont to find the candidates with desired resume.

The scope of future work can affect incremental learning, which stores the prevailing model and processes the new incoming data more efficiently. More specifically, the models with incremental learning are often utilized in categorization process to enhance the subsequent aspects in each sort of problems.

Project are often extended to a huge area because every company or institute needs the documented data to be organized and therefore the documented data which must be organized is in huge amount and remains increasing quantitatively. Combining with different languages different parts of the world could make precise decisions of selection, while working with bulk of documents. Better classified and arranged file helps in continuous work flow with none interruption of selecting and organizing files.

REFERENCES

[1]. Sang-Woon Kim and Joon-Min Gil "Research paper classification systems based on TF-IDF and LDA schemes," Kim and Gil Hum. Cent. Computer Inf. Sci.,2019.

[2]. Wei Li and Andrew McCallum "Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations," 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

[3]. Md.Saiful Islam, FazlaElahiMdJubayer and Syed Ikhtiar Ahmed "A Support Vector Machine mixed with TFIDF Algorithm to categorize Bengali Document,"International Conference on Electrical, Computer and Communication Engineering (ECCE), 2017.

[4]. AnkitaDhar, NiladriSekhar Dash and Kaushik Roy "Categorization of Bangla Web Text Documents Based on TF-IDF-ICF Text Analysis Scheme," 52nd Annual convention of the Computer Society of India, CSI 2017.

[5]. MowfyM,Rezk k and El-bakry HM "An Efficient Classification Model For Unstructured Text Document,"American Journal of Computer Science and Information technology , 2018.

[6]. PuneetGoswami and Vidya Kamath "The DF-ICF Algorithm- Modified TF-IDF," International Journal of Computer Applications (0975 – 8887) Volume 93 – No.13, May 2014.

[7]. Ms.SharmilaShinde, Dr.PrasannaJoeg and Dr. Sandeep Vanjale "Web Document Classification using Support Vector Machine," International Conference on Current Trends in Computer, Electrical, Electronics and Communication, 2017.

[8]. KanikaDhanjal and SangitaBhagat "Applying Machine Learning Algorithms for News Articles Categorization: Using SVM and kNN with TF-IDF Approach,"A. K. Luhach et al. (eds.), Smart Computational Strategies: Theoretical and Practical Aspects,https://doi.org/10.1007/978-981-13-6295-8_9,2019.

[9]. Manjula S. and Ravindra S. Hegadi "Identification and Classification of multilingual document using maximized mutual information," International Conference on Energy, Communication, Data Analytics and Soft Computing, 2017.

[10]. Mingyong Liu and Jiangang Yang "An improvement of TFIDF weighting in text categorization,"International Conference on Computer Technology and Science, 2012.

Rajnish Virpate, et. al. "Document Classification using Machine Learning Techniques." *International Journal of Engineering Research And Development*, vol. 16(4), 2020, pp 48-55.