DOCUMENT CLASSIFICATION USING MACHINE LEARNING

Shipra Trivedi#1, Komal Malawat#2, Namrata Yerawar#3, Shivani Chaudhary#4, Asmita Kamble#5

Department of Computer Engineering, First-Third University

[1]shipratrivedi1998@gmail.com

[2]komalmalawat1998@gmail.com

[3]namrata.yerawar1@gmail.com

[4]shivanichaudhary2016@gmail.com

Abstract

Document classification is a problem in information and computer science. It is basically the process of categorizing documents in certain categories correctly. It is considered as one of the key techniques used for organizing the data by automatically assigning a set of documents into predefined categories based on their content. Recent advances in computer and technology resulted into ever increasing set of documents. The need is to classify the set of documents according to the type. So, the classification is widely used to classify the text into different classes. This paper proposes a document classification system to identify the domain of the document. This classification is going to be performed by using Naive-Bayes approach which is one of the machine learning algorithm. It consists of a set of phases and each phase can be accomplished using various techniques. Selecting the proper technique that should be used in each phase affects the efficiency of the text classification performance.

Keyword : Naïve Bayes (NB), Machine Learning (ML), Natural Language processing (NLP), Term-Frequency inverse document frequency (TF -IDF)

I. INTRODUCTION

Document classification is the task of grouping documents into categories based upon their content. Document classification is a significant learning problem that is at the core of many information management and retrieval tasks. It performs an essential role in various applications that deals with organizing, classifying, searching and concisely representing a significant amount of information. This is especially useful for publishers, news sites, bloggers or anyone who deals with a lot of content like managing growing repositories of documents in an organization. By clustering and categorizing documents, the work done in these areas can be completed easily. The size and number of online and offline documents is increasing exponentially. The need for identifying groups of similar documents has also increased for either getting rid of multiple versions of same documents or extracting relevant set of documents from huge document repositories.

This paper uses machine learning techniques for classification of documents. Machine Learning enables systems to recognize patterns on the basis of existing algorithms and data sets. Machine Learning undoubtedly helps people to work more creatively and efficiently. Therefore, in Machine Learning, artificial knowledge is generated on the basis of experience. In this system, by classifying document, one or more categories are assigned 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.

Machine Learning uses different algorithms to train systems like Support Vector Machine (SVM), Naïve Bayes, K-nearest neighbour (KNN), Decision tree, K-means etc. In this paper, Naïve Bayes algorithm is being used as by comparing different algorithms, Naïve Bayes shows highest efficiency and it is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

The rest of this paper is as follows. We summarize the reference paper and brief description about the study is mentioned in section II. Section III discuss the use of various algorithms like Naïve Bayes, TF-IDF, etc. Scope of the project is discussed in section IV. Finally, we conclude the paper in section V.

II. Related Work

Nowadays finding the papers related to a particular domain is very difficult until we read the whole paper. Classification and summarization is a machine learning technique that assigns categories to a collection of data to aid in more accurate predictions and analysis. The ideas about classification based on experimental results are discussed in [10]. The classification problem is one of the most fundamental problems in the machine learning and data mining literature. In the context of text data, the problem can also be considered similar to that of classification of discrete set-valued attributes, when the frequencies of the words are ignored. Therefore, text mining techniques need to be designed to effectively manage large numbers of elements with varying frequencies are stated in paper [9]. Text and document classification processes are often used in areas such as sentiment analysis, text summarization, etc. The author Mehmet Baygin [1] has performed document classification using Naive Bayes approach. Preprocessing steps were applied to the dataset, the stop words and punctuation marks were cleared from the dataset and then, N-gram feature extraction for each document was performed.

After look over the papers, the author Krina Vasa[6] stated that there are so many techniques in text classification. Support vector machine, k-nearest neighbor and nave Bayesian method are widely used techniques in text classification. The hybrid approach of these techniques also very useful in text classification. Text and document classification processes are often used in areas such as sentiment analysis, text summarization. Documents need to be clustered and categorized in a quite successful way so that the work done in these areas can be successfully completed. The proposed approach has been tested on a real dataset and achieved a performance of about 92%. And this approach is highly efficient and classifies the documents with great accuracy.

The process of classification algorithms based on text and their comparison based on data size, type of data, transparency and the situations in which each algorithm works better is discussed in [4]. The text classification algorithms used in this paper are Decision Tree, Naive Bayes, K-nearest neighbour and Support Vector Machine. This paper has made a study of algorithms on different data set. Decision Tree is best for larger samples and Naive Bayes, KNN and SVM work best when the sample size is small. This paper has provided an insight about when to use which classification algorithm assuming that the information of the data set is fully understood. The author formulates the best criteria under which each algorithm performs better. This might help analysts to choose a better classification algorithm. Also in paper [7], the author Muhammad Rafi et al compared SVM and naïve bayes results and found that naïve bayes is better.

[2] states that clustering techniques can be applied only on the structured data, so unstructured data need to be converted into structured data**.** But while converting unstructured data into structured data the algorithm efficiency decreases, so to increase the efficiency we need to reduce the number of terms as they are more in number. It expects to extract the terms from the documents, so documents represent rows and terms are placed in columns. The terms are in large number which causes the

problem of dimension curse and decreases algorithm efficiency. For this, the author has used TF-IDF approach. TF-IDF technique is used to eliminate the most common terms and extracts only most relevant terms from the corpus.

Recently, numerous research activities have been conducted in the field of document classification. Document classification is a growing interest in the research of text mining. Correctly identifying the documents into particular category is still presenting challenge because of large and vast amount of features in the dataset. In this paper, the author S.L. Ting et al [5] has highlighted the performance of employing Naïve Bayes in document classification. Results show that Naïve Bayes is the best classifiers against several common classifiers such as decision tree, neural network, support vector machines, KNN, Naïve Bayes, in term of accuracy and computational efficiency. Among these classifiers, the Naïve Bayes text classifier has been widely used because of its simplicity in both the training and classifying stage. Naïve Bayes models allow each attribute to contribute towards the final decision equally and independently from other attributes. In the paper [8], the usefulness of that approach have been presented using the probability of various abstract papers and obtained the satisfactory results in term of accuracies.

Table No. 1: Algorithm used and their features

Paper Name	Algorithms used	Main Context	Paper Features	Weakness	Strengths
Classification of Text Document based on Naïve Bayes using N-Gram Features	Naïve bayes	Document classification is done using Naive Bayes approach.	Tested on a real data set Accuracy-92% Highly efficient	Very sensitive to the form of input data.	Highly efficient More accurate
Document Clustering : TF-IDF approach	TF-IDF	To increase the efficiency, TF-IDF approach has been used.	Eliminates the most common terms and extracts only most relevant terms Based on bag-of-words model.	Works poorly with unstructured data. As it is based on bag of words model, it is only useful as a lexical level feature.	Improves efficiency of structured data. Simple to execute
An outcome based comparative study	Decision tree	Provides the impact of various text classification under different	Decision tree - features have to be checked in a specific order.	SVM is not suitable for large dataset. In decision trees,	SVM is relatively more efficient. Decision trees

of different text classification algorithms	SVM	scenarios. Also provides an insight about when to use which classification algorithm.	Many improvements and modifications done to SVM.	calculations get very complex particularly if many values are uncertain or are linked.	can be combined with other decision techniques.
An Efficient Classification Model for Unstructered Text Document	TF-IDF Naïve Bayes	Classification model that supports both the generality and the efficiency.	Use of Multinomial Naive Bayes with TFIDF is more superior approach. Presents the logical sequence of text document classification	TF-IDF can easily compute similarity between two documents. Very sensitive to the form of input data.	TF-IDF can easily compute similarity between two documents. Naïve bayes is highly efficient
Is Naïve Bayes a Good Classifier for Document Classification ?	Naïve bayes	Simplicity in both the training and classifying stage.	Allows each attribute to contribute towards the final decision equally More computational efficient Best document classifier.	Very sensitive to the form of input data.	Naïve bayes requires a small amount of training data to estimate the test data.

III. PROPOSED SYSTEM

This paper proposes a document classification system to identify the domain of the document depending upon the keywords. Classification is going to be performed by using Naive-Bayes approach which is one of the machine learning algorithm. It accepts input as PDF documents that will be classified based on their domain. By extracting keywords from input PDF, the system will identify the domain of the paper. It classifies the given input documents into limited number of domains only. This system will also generate summary of keywords that belongs to the particular paper as per their domain.

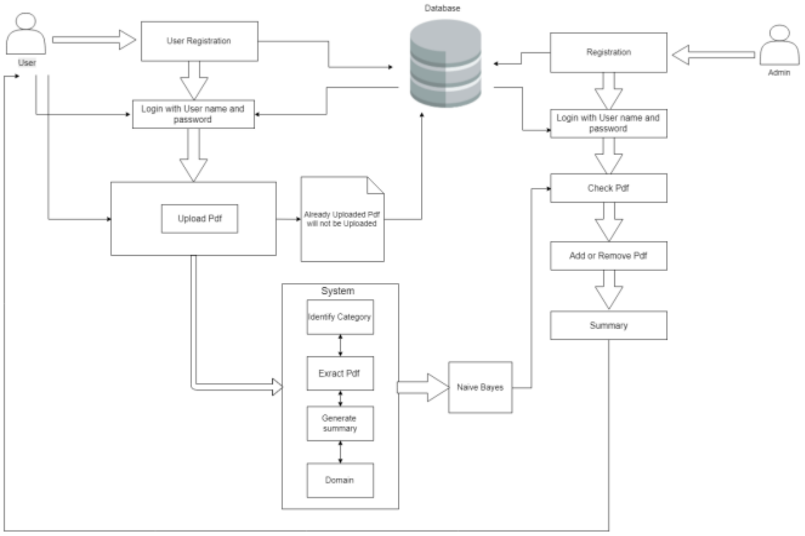


Fig. 3.1 Architecture Diagram

In the figure 3.1, the system architecture describes the overall flow of the system. This system is useful for the early classification of the post. The user who will use this system needs to first register into the system. The details will be stored in the database. After registration, the user will log in to the system using the login page. The algorithm used in the system is Core NLP for text mining i.e. for removing stop words, after mining TFIDF is used for extracting main words and Naïve bayes is used for classification of documents based on their content.

There are basically two main approach for classification of documents:

A. ***Extract keywords:***

Firstly the PDF will be uploaded then keywords are extracted and compared with the keywords that are stored in the database, for this process TF-IDF algorithm has been used. TF-IDF (Term Frequency –Inverse Document Frequency) is used to convert a document into structured format. It is a numerical to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval. The TF-IDF value increases proportionally to the number of times a word appears in the document. And thus the main words are extracted from the pdf.

TF-IDF:
Step1: Clean data / preprocessing - Clean data(standardise data),Normalize data(all lower case),lemmatize data(all words to root words).
Step2: Tokenize words with frequency
Step3: Find TF for words
Step4: Find IDF for words
Step5: Vectorize vocab

TF: Term Frequency, which measures how frequently a term occurs in a document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. Thus, the term frequency is often divided by the document length (aka.

the total number of terms in the document) as a way of normalization:

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

IDF: Inverse Document Frequency, which measures how important a term is. While computing TF, all terms are considered equally important. However it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus we need to weigh down the frequent terms while scale up the rare ones, by computing the following:

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

Example:

Consider a document containing 100 words wherein the word *cat* appears 3 times. The term frequency (i.e., tf) for *cat* is then (3 / 100) = 0.03. Now, assume we have 10 million documents and the word *cat* appears in one thousand of these. Then, the inverse document frequency (i.e., idf) is calculated as log(10,000,000 / 1,000) = 4. Thus, the Tf-idf weight is the product of these quantities: 0.03 * 4 = 0.12.

B. *Identify domain:*

The PDF will be checked in the system to find it’s domain. For this purpose, the important keywords are extracted. And then the extracted keywords are matched with keywords stored in the database. Now, the classification of documents is done using Naïve Bayes approach. Naive Bayes is a simple classification method based on the Bayes theorem. In this Bayes rule applied to documents and classes and for this conditional probability is calculated. If domain is not identified, the system will categorize that PDF as other category.

NAÏVE BAYES :

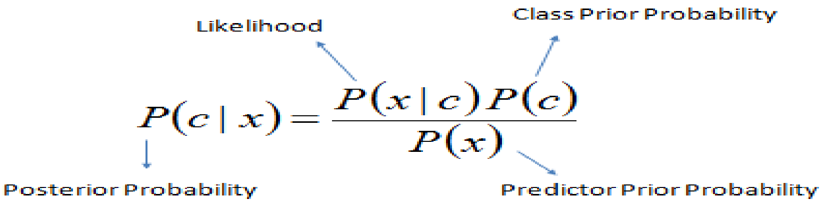
Naive Bayes classifier calculates the probability of an event in the following steps:

Step 1: Calculate the prior probability for given class labels

Step 2: Find Likelihood probability with each attribute for each class

Step 3: Put these value in Bayes Formula and calculate posterior probability.

Step 4: See which class has a higher probability, given the input belongs to the higher probability class.


Likelihood
Class Prior Probability
$P(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)}$
Posterior Probability
Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

IV.RESULTS AND DISCUSSION

Document classification is basically the process of categorizing documents in certain categories correctly. By classifying documents, we are aiming to assign one or more classes to a document making it easier to manage and sort. Recent advances in computer and technology resulted into ever increasing set of documents is the need to classify the set of documents according to the type. This project proposes a document classification system to identify the domain of the document depending upon the keywords. Classification is going to be performed by using Naive-Bayes approach which is one of the machine learning algorithm .

For classifying any document, user must be logged into the system using a username and a password and after entering the correct credentials, he/she is allowed to log into the system. After the successful login, user have to upload the document into PDF format only. Now the keywords from the document are extracted using TF-IDF(Term Frequency-Inverse Document Frequency) and are compared with the keywords that are stored in the database. TF-IDF is a numerical to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval. It eliminates the most common terms and extracts only the most relevant terms. Now the document is classified using Navie-Bayes algorithm. Naive Bayes acted as a base for text classification which is easy and works faster. It is a simple classification method which relies upon on Bayesian rule with strong independent assumptions among the features. Bag of words are the simple representations of a document. If the domain not identified, the system will categorize that PDF as other category.

Finally, after the domain is identified, a short summary of the keyword is generated. This system uses machine learning techniques for classification of documents. Machine Learning enables systems to recognize patterns on the basis of existing algorithms and data sets. Machine Learning undoubtedly helps people to work more creatively and efficiently. Therefore, in Machine Learning, artificial knowledge is generated on the basis of experience. This is especially useful for publishers, news sites, bloggers or anyone who deals with a lot of content like managing growing repositories of documents in an organization.

By clustering and categorizing documents, the work done in these areas can be completed easily**.**

V.FUTURE WORK

The system proposed in this paper work only for the documents of PDF format using machine learning. In future, the scope of this system can be extended to other formats than PDF. With the increase in development of different machine learning algorithms, the system will be able to identify all the domains if the input document is a combination of more than one domain.

VI. CONCLUSION

The size and number of online and offline documents is increasing exponentially. The need for identifying groups of similar documents has also increased for either getting rid of multiple versions of same documents or extracting relevant set of documents from huge document repositories. So, this paper helps us to classify the documents into different categories according to their content. In this paper, a Document Classification System for classifying the research paper into different categories

(Machine Learning, Artificial Intelligence, IOT, etc...) have been proposed. By using this, we can find the domain of our document easily. This is especially useful for publishers, news sites, bloggers or anyone who deals with a lot of content like managing growing repositories of documents in an organization. It applies the naive Bayes algorithms to classify documents automatically. This classifier gives a correct and accurate result when compared with other Machine Learning classifiers.

REFERENCES

[1] Aggarwal, C. C., & Zhai, C, "A survey of text classification algorithms" , (2012).

[2] Mamoun, R., & Ahmed, M. A, "A Comparative Study on Different Types of Approaches to the Arabic text classification". In Proceedings of the 1st International Conference of Recent Trends in Information, (2016).

[3] A.Helen Victoria, M.Vijayalakshmi, "An Outcome-based Comparative study of different Text Classification Algorithm". Volume 118 No. 22 , 1871-1877, (2018).

[4] Ahmed, H.A. & Esrra, H. A. A, "Comparative Study of Five Text Classification Algorithms with their Improvements". International Journal of Applied Engineering Research, 12(14),4309-4319, (2017).

[5] Badgujar, M. G. V., & Sawant, K., ``Improved C4. 5 Decision Tree Classifier Algorithms for Analysis of Data Mining Application". International Journal, 1(8), (2016).

[6] Krina Vasa , "Text Classification through Statistical and Machine Learning Methods." Volume 4, Issue 2, ISSN: 2321-9939,(2017).

[7] Sundus Hassan, Muhammad Rafi, Muhammad Shahid Shaikh , "Comparing SVM and Naïve Bayes Classifiers for Text Categorization with Wikitology as knowledge enrichment." , (2017).

[8] Anuradha Purohit, Deepika Atre, Payal Jaswani, Priyanshi Asawara , "Text Classification in Data Mining" International Journal of Scientific and Research Publications, Volume 5, Issue 6, ISSN:2250-3153.(2015).

[9] Charu C. Aggarwal, Cheng Xiang Zhai , "A Survey of Text Classification Algorithm" , (2018).

[10] Zun Hlaing Moc, Mic Mic Khin, Thinda San, Hlaing May Tin , "Comparision of Naïve Bayes and SVM Classifiers on Document Classification". 7th Global Conference on Consumer Electronics, (2018).