International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 15 (2018) pp. 12388-12395

Supervised Learning Based System for Classification of Wikipedia Articles

Raghavendra S[1], Dr. Lingaraju G M[2], Shekar sivasubramanian[3]

[1, 2] *Dept of Information Science and Engineering, Ramaiah Institute of Technology, Bengaluru, Karnataka, India.*

[3]*Researcher, Carnegie Mellon University (CMU), Language Technologies Institute, USA.*

Abstract

Web is a huge source of unstructured information, which is not so useful for human beings until it is classified. Text categorization is the process of classifying these unstructured documents into predefined labels based on the content. Here automated text categorization takes input as large set of text documents and group them into predefined labels such as sports, politics, entertainment etc. This can be useful for multiple domains. In the existing situation users classify the documents manually by reading the content of the document, but find one approach followed is where a high degree of effort is required for reading large set of text documents. To classify the large set of text documents user need to read all the documents which is time consuming and costly process. In order to increase the efficiency of classification, it should need more man power to classify the document for large set of text documents. In order to solve the problems faced by the existing system, a novel automated text categorization based system is proposed for classification of documents on Wikipedia articles using supervised machine learning and measure the accuracy of supervised learning algorithm. The experiment is conducted over 564 documents of Wikipedia articles. The accuracy of Support Vector Machine and K Nearest Neighbor are obtained by the proposed system in classifying the documents are 88% and 85% respectively.

Keywords: Text categorization, Support Vector Machine, K Nearest Neighbor, Supervised machine learning, TFIDF.

INTRODUCTION

Technology is exponentially increasing in the existing real time world. So, storing data is interdependent of the technology. Now days the users are actually transaction with the text formatted data rather than numerical oriented data. The different technology used in the real time world is sometime difficult to apply on the textual data. For example text classification is one among the technologies to work on the text formatted data. Therefore it's very crucial to deal a technique which is related to textual data which is completely different from the numerical data. Text classification is one which deals with the mining of textual data. Some of the basic pattern which is interrelated to the text classification was rules and synthesizing and analyzing the relations[15]. Based on the some experimentation the information can be extracted based on the new facts. In order to find the relevant information the problem arises in this context is not relevant to the user needs, so the ultimate goal of text classification is to discover the information which is not known or which does not exist. Some of the functions[16] of the text classification is the summarization of text, clustering of text and categorization of text.

Text categorization is the process of grouping documents into a set of categories based on their content[12]. Text categorization is an important learning technique that is at the core of many tasks such as management of information and information retrieval. Text categorization performs a prominent role in different applications that deals with organizing the documents, classifying the documents and concisely representing a vast amount of text documents. Text categorization is a well established problem in information retrieval. Automatic text categorization is the process of automatically classifying the relevant category of new documents from the testing set of Wikipedia text documents using supervised learning. In Supervised text categorization, some preprocessing tasks such as tokenization of text, eliminating stop words, stemming of text and select the features of text documents. Then train the documents with class labeled text documents known as training dataset into the classification model. This model provides information related to the correct categorization of documents against testing dataset with no class labeled text documents. So that in case of supervised learning technique, it becomes easy to test the accuracy of text categorization model.

In the digitalizing world the increasing growth of electronic documents and accessing information is difficult, so predominantly its conflict on efficient and effective manner for accessing the information for the document. Furthermore, to productize the information the categorization of text is a part of enhancement. In the real time digital documents the text classification is been a research topic and imperative application. So the classification of text is persistently based on the huge amount of text formatted data. Text categorization is one in which the category of document is based on the predefined content of the document[10]. To manage a huge amount of text formatted documents, document classification plays a prominent role. Single label and multi label are the two types of text categorization. The class or one group is essentially appropriate and is required to assign each document is called as single label[11]. Multi label text grouping is one in which one or multiple groups can be empowering to a document. The challenging task in text classification and it mainly has two main factors that are: feature extraction and classifying documents.

The main set of feature that accurately describes the document is obtained from the feature extraction and it also builds best classification model. Good model can be obtained for classification from feature extraction. The documents which contain the broad topics are very complicated and categorizing then is most difficult task. When the document which contains information about theocracy is considered such document makes it very difficult to categorize it has politics and religion. The document may contain the topic which is broad and it contains different meaning based on the context which might appear multiple times within the document with different context in the document.

The organization of the paper is as follows: A Brief introduction about the paper is discussed in Section I. We cover the study of the Literature Review that are already done in this field and are considered as the base study for this work discussed in Section II. In Section III, describes the Proposed Method system architecture. Results and Discussion inferred from the output is discussed Section IV. Section V contains the conclusion of the Paper.

LITERATURE REVIEW

W. Zhang et al, Used that support vector machine which is building block for the text classification of multi-word. Author used multi-word phrases, a syntactical structure for multi word extraction. Author used pattern identification for reducing the computation cost and proposed the extracted duplicate patterns for concatenating the regular expression for the multiple words[1]. Based on the semantic levels of the multi-words, they established two methods based on the different strategy. The author suggests improving the performance of the system by using semi-supervised learning.

Text visualization which is embedded with the word clouds for text analytics. Here they extracted data from the news articles. The text is converted into clouds of words based on the information and features which works interactively on the word cloud explorer[2]. Author addresses the issues related with comparison and handling of multiple documents and also suggests integrating the cloud with the present approach to improve the performance.

Support vector machine on the sport articles to perform the text classification. By using the support vector machine classifier the author provided grouping system which categorizes the relevant text. Based on ranking of support vector machine it performs the multi classification system. Here [3] proposed SVM light tool which classifies sports articles as sport relevant. The author suggests using SVM ranking in the future scope.

Hybrid text classification with naive bayes and support vector machine. Here the dataset is initially split into testing and training dataset. The input data to this support vector machine is in the form of numerical values. So testing and training data are transformed into numerical values using Bayesian vectorization method. Then this numerical data are inputs for support vector machine classifier and then output of this classified data can be used as a training dataset [4]. Author suggested increasing performance of the system using artificial intelligence machine learning.

Text and document classification using machine learning with KNN classifier. Here Text classification is the difficult task generally suitable credential into the predefined classes. Single label categorization is one in which the text resides to one association. Multi label categorization is one which includes multiple documents. The author actually classifies the credentials with the help of KNN based on the machine learning approach. Comparison of naive bayes and term-graph methods by returning more precise documents are given in [5]. The major drawback of using KNN is time consuming but it provides the better correctness as compared to others. The mixture of these classifiers provides the better outcome than the combination of other.

The challenges of text categorization in twitter data set using machine learning techniques. They used machine learning techniques that depend on an inclusive set of features derived from user information. In the outcome was experimented on 3 tasks with different attributes like detecting the political affiliation, ethnicity identification and detecting affinity for particular business [6].

Mita K. Dalal et al. deals with some big issues like unformatted data. In author also describe the automatic text document. Some of the major application of automatic text categorization is opinion mining, contextual search and product review process and text sentiment analysis [7]. Author suggests improving the performance using advanced text mining techniques.

PROPOSED SYSTEM ARCHITECTURE

The proposed system architecture aims in the design and development of an automated system for classification of text document on Wikipedia articles using supervised machine learning and classify the label for the new document.

In system architecture, the extraction of the text documents from the downloaded xml dump file which contains the Wikipedia articles using the python packages and the lxml. Extracted text documents are in unclean form need to clean the text documents using basic regular expressions methods and natural language processing methods. Natural language tool kit(NLTK) performs the preprocessing tasks on the extracted text documents to get the clean text documents for building the corpus. Tokenizing the text, removing the stop words and then stemming the text are the preprocessing techniques on the text documents for removing the unnecessary text and the required features for training the algorithm. Prepare the training dataset and the test set from corpus for training and testing the algorithm. Features are extracted from the text documents of corpus and select the features from the training dataset and the test dataset of text documents are transformed into numerical vector form using tf-idf method for training and testing the algorithm. Prepare the label data for training the model with the training dataset in supervised learning. Documents are classified the category based on the content or extracted features using support vector machine algorithm. Finally classify the label for the new document for test set from trained algorithm using the sklearn

of python package as described in the below Figure 1.

Figure 1: System Architecture.

Data Collection and Cleaning

In text categorization, the users provides a dataset of text documents from Wikipedia articles and uses a part of dataset for training the classifier model and then classify the result of the remaining text documents. The dataset of text documents are organized into a class. The method of training text documents from a set of correctly classified text documents is known as "Supervised Learning". Actually for training the model each text document is giving with labeled data. The trained datasets with labeled data are used for classifying the correct class for new set of unlabeled or unclassified text documents. In this work we considered the correctly classified text documents dataset from Wikipedia articles.

Generally the Wikipedia articles are just text or paragraphs. But this set of labeled text documents consists of documents for each articles and each document is provided with class label. This process of the data processing in text documents a bit hard with this unwanted text. The text documents from Wikipedia articles can contain special characters, images and some other noisy data. These text documents should be cleaned to reduce all those junk data in the text file to be processed in following stage.

Building Corpus

In the previous stage, we have prepared the data, and create the corpus and store text documents in the text corpus. We have to organize and divide the datasets into training dataset and testing dataset. The training dataset can be used for train the model and the testing dataset can be used for testing the data. In machine learning while using the supervised learning, it is required to train the model with the use of the training dataset along with the labeled data in machine learning. This training dataset is required for the entire dataset to increase the accuracy of the classification model. In most of the machine learning preprocessing techniques are separated into the train datasets and test datasets are using 80-20 rule, where 80% of dataset is used for training the model and 20% of the dataset is used for test dataset.

Data Pre-processing

Documents are given in the form of unstructured text data which usually requires a transformation of text into the representation of machine process-able format. Eliminating stop words and stemming of the text are the important part of preprocessing phase for cleaning the text from the text documents[19]. So the process of conversion of the machine readable format from the raw text documents is very much required. So that this process of transforming the text documents into a representation of number format, which are suitable for the need of learning algorithm. This phase deals with the text categorization techniques of encoding documents for text classification tasks. In text categorization, the representations of numerical vectors are encoded from the text documents. This process of converting is done in two parts. The first part of the encoding process deals with the extraction of text features from the article corpus and the process of selecting some of the features from the corpus. The second part of the encoding process provides corresponding values to the features this last step can make for features generating numerical vectors.

Feature Extraction

Feature extraction is the one of the text preprocessing step which is used to reduce features which are repeated. So the text preprocessing tasks such as removal stopwords and stemming the text are performed on the text documents to reduce the unwanted features[19]. The representation of documents in text categorization which contains the large number of features and some of the features are irrelevant or noisy[20]. The two main purposes of the feature extractions are. First, the extracted text features makes increases the efficiency of the classifier with the training dataset for categorize the new documents by reducing the effectiveness of the text vocabulary. Second, selecting the features will be frequently increases the accuracy of the classification model by cleaning the noisy data or unnecessary text feature that is not required for the prediction of the classification model. These noisy text features or unnecessary words are present in the document representation, this will also increases the classification error on new data. So that unnecessary words or text features are reduced in the text documents for text categorization by natural language processing methods of the text preprocessing tasks are eliminating the stop words and stemming the text. These steps of extraction of text features are given below.

Tokenize the text

Tokenize the text is the process of making or breaking the meaningful words or phrases from the sentences. The achieving tokenization method by using the separation of comma, stop and space. The python built in nltk libraries can tokenize the text documents can split into words and convert all the text documents into lowercase.

Eliminating stop words

When working with the text categorization techniques, eliminating the stop words is the standard method to eliminate the irrelevant or unnecessary words or text from the features of text documents. If we remove the stop words from the documents, we will decrease the number of text features from the corpus and instead of concentrating other unnecessary feature. We can concentrate more on the actual required feature for training the classification model.

Single set of words are known as stop words. For example determiners such as 'a, and, the', prepositions such as 'up, down, under'. These type stop words which will be removing from the text corpus. Here we are mainly considering the Wikipedia articles.

The python built in nltk library is used to eliminate the stop words from the text documents.

Stemming the text

Stemming the text is the process of trimming the text into the original form of word. For example, if we have a word 'training' and 'trained' stemming the text will cut off that word to 'train' and then use that same text feature for the occurrences of those words.

Vector representation

The original forms of text documents are unsuitable for the learning of the algorithm. These text documents are required to convert into the required input number format for training or learning the algorithm. Therefore the attribute – value representation of documents are needed for the learning of algorithm, which means the converting text documents into vector space.

Once the extraction of text documents, these documents needs to be preprocessed using the preprocessing techniques such as tokenizing the text for breaking the sentences into words, stop words removal for reducing the repeated words from the text documents and then stemming the text is for cutting off the text into distinct words, etc. To select the features from the corpus documents is important motivation for feature selection. The predetermined importance of the word is measured according to keeping the terms with highest score is performed by feature selection[20].So the transformation of the documents into vector form can be happen. The corresponding each term can have the one dimension in the document, the identical terms can also contain similar dimension. Then the term i equivalent to the j^{th} dimension of vector space. The standard technique is for conversion of the text documents into a vector form and the term weighting technique [8] is known as TF-IDF. The TF-IDF weighting method is very important for building model for understanding and learning the algorithms like SVM and KNN.

The TF-IDF is also known as Term frequency – Inverse document frequency. The information and the text classification techniques are used TF-IDF for the conversion of documents. This TF-IDF weight is statistical measure required to decide how significant a term for the document from the collection of documents or text corpus. The importance rises comparable to the number of times a term can takes place in the document over the frequency of the term from the set of corpus. IDF is from the text corpus measures how the term is impromptu in the corpus. Hence if the word exists frequently in the text corpus and then it is not considered for training the document. If the word is does not exists frequently in the text corpus then it is considered for the relevant for the document.

There two terms in the composition of TF-IDF. They are one is to evaluates the normalized Term frequency (TF) is nothing but the number of times the term exists in a document, separated by the whole number of terms from that document and then the next one is Inverse document frequency (IDF) method is to evaluates the logarithm of the number of documents from the corpus separated by the number of documents containing the word takes place. The TF-IDF denotes as below.

$$TF\text{-}IDF = TFeq \times IDocFeq$$

To compute the Term Frequency(TFeq) is

$$TFeq(i, j) = TFeq(i, j) \,.\, DocFeq(i)$$

And then to compute the Inverse Document Frequency(IDocFeq) is

$$IDocFeq(i) = \log(N/DocFeq(i))$$

Where, TFeq(i, j) is the number of times the word i hold in the document j.

N is the total number of documents in the corpus.

DocFeq(i) is the number of documents hold the word i.

Classification algorithm

Support Vector Machine: SVM is one of the most frequently used algorithm, which divides a single input into two different sets which includes training and testing set. For instance, a document can be classified into special and on-favored sets. Based on training set documents SVM aims to discriminate between two categories which include the labeled documents in the two categories. The internal working of SVM is based on kernels like linear and non-linear is rbf, such that it handles the documents so that in a high dimensional space it can be shown as points which then could be used to find a hyper-plane that ideally isolates between the two categories.

Learning and classification of documents based on the extracted features of the text documents are classified and classify the label for the new documents by using support vector machine classification algorithm. SVM constructs a hyper plane in a high dimensional space[17] for the linear kernels. Accurate separation is achieved by the hyperplane, if it has largest distance to the nearest training data point using kernels[18]. The classification approach of the system

randomly divides the dataset into training and testing dataset. The proposed system is to classify documents using support vector machine classification algorithm. SVM is one of the advanced trend in machine learning used for solving many text classification problems. SVM mainly deals with two classes by maximizing or minimizing the margin from the hyperplane as shown in figure 2. The object samples close to the margin that are used to define the hyper plane are called as Support vectors.

Depending on the way the given points are separated by hyperplane, the SVMs can be classified into linear SVM and non-linear SVM. Hyperplane is defined by positive and negative values. The mathematical formula for finding hyperplane is given by

$$(a.b) + c = +1 \ (positive\ label)$$

$$(a.b) + c = -1 \ (negative\ label)$$

$$(a.b) + c = 0 \quad (h\ yper\ plane)$$

Where, a, b and c are individual classes defining the hyperplane. The values of these classes are calculated by using linear algebra.

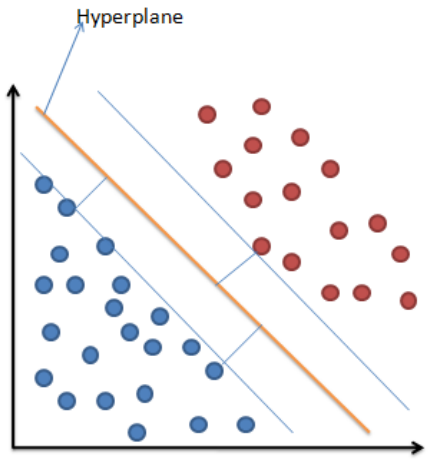

Figure 2: Optimal separating hyperplane of SVM

The proposed system uses single label classification to classify 7 different classes. Single label classification can be done by one class SVMs by taking one to one combinations. For classification, a training data set of 80% text documents and testing dataset of 20% text documents is taken into consideration. Classification is performed on the trained data set consisting seven classes, namely to Sports, Language, Cities, Science, Technical, Politics and Entertainment. Each of the classes has equal or different size of text documents and defined by their own specific features.

K Nearest Neighbor: K-NN algorithm is an instance based algorithms which uses the instances of data to perform the predictions.

Euclidian Distance:

Euclidian distance is used in the NN algorithm when this is dense data or continuous data. The distance can be used as the proximity degree.

The steps followed are:

- Measure "dist (a, a_i)" i =1, 2, ….., n; where dist is the Euclidian_Distance between the 2 points.
- Organize the measured n Euclidean_distances in increasing order.
- Let x be the positive integer, take the first x distances from the list which is sorted.
- Discover y-points conforming to x-distances.
- Let xi signifies the number of points fitting to the ith group amongst x points i.e. $x \geq 0$

  If $x_i > x_j \ \forall\ i \neq j$ then put y in class i.

Model Testing

Many evaluation techniques compare the labels for the input data and calculate the score. These inputs will have same format as the training data set. We have chosen a set of testing documents from Wikipedia articles to evaluate the performance of the implemented system. These testing documents are different from the training data sets. Using the same training dataset for testing would simply memorize scores and results in misleading scores. We have used precision, accuracy, F-Score, recall metrics to calculate weather labels are classified correctly.

RESULTS AND DISCUSSION

Accuracy Evaluation

The successful application in industries are essential with success of algorithm is measured based on the accuracy or the predicted accuracy of the algorithm. In recent years many testing procedures are built to compute the performance of the classification algorithms. The process of measure the outcome of the automatic classification how near to match the correct categories of the documents is known as accuracy. To determine the accuracy is that the user needs to classify the testing dataset by applying the classifier for the dataset. The user given the documents for classification algorithm is to classify the document that can be trained with the labels which are classified correctly as given same label assigned by the user that are known as the correctly classified documents and the remaining documents are misclassified or incorrectly classified.

The accuracy is the evaluation of number of correctly classified documents over the sum of the number of correctly classified documents and incorrectly classified documents.

Therefore, accuracy = (number of correctly classified documents) / (number of correctly classified documents + incorrectly classified documents)

The training classifier will not use to ensure the documents of the testing set for training that is needed for determine the accuracy of the model. This make sure that the classifier will not have suitable information from the testing documents of the corpus that will highly increases the performance of the model. Generally, the corpus is split into training set and testing set and then prepares the labeled documents for training the algorithm. The training set of documents and the labeled data are used to train the algorithm and then testing set is used to apply for testing the algorithm to determine the accuracy. Hence any of testing documents will not come into the training set. The performance of the classifier will be the best estimator of predicted accuracy on obscured data.

The evaluation accuracy of the classifier rely on four features [9]. They are as follows-

a) True positives(TP) are the correctly classified which are relevant documents as per the proper label.

b) True negatives(TN) are the correctly classified which are irrelevant documents as an improper label.

c) False positives(FP) are the incorrectly classified which are irrelevant documents as proper label. Hence it is type I error.

d) False negatives(FN) are the incorrectly classified which are relevant documents as improper label. Hence it is type II error.

The evaluating accuracy for two parameters are from these four features, i.e. F1 score is computed. The two parameters are Precision and Recall.

- Precision is the one of the parameter which specifies as proper label how many number of documents are classified.

  i.e. Precision = TP/(sum of TP and FP)

- Recall is the another parameter which specifies how many number of relevant documents which are classified.

  i.e. Recall = TP/(sum of TP and FN)

- F1 score is the harmonic mean of precision and recall is denoted as-

  i.e. F1 score = 2*(the product of Precision and Recall)/(sum of Precision and Recall)

The proposed method is implemented for 564 Wikipedia articles. 80% of Wikipedia articles are used as training dataset to train the algorithm. Training dataset consists of Wikipedia articles which are related to sports, Language, Cities, Science, Technical, Politics and Entertainment. Based on the training dataset, SVM scikit learn build the model for prediction. 20% of Wikipedia articles are used as the testing set for testing the classifier which is related to sports, Language, Cities, Science, Technical, Politics and Entertainment. We are planning to increase the size of dataset.

Table 1: represents the comparison of performance evaluation in terms of accuracy, precision, recall and f1-score. The result can be compared by using Support Vector Machine and K Nearest Neighbors classifier. The evaluation results of the different kernel functions of SVM(support vector machine) and the K nearest neighbor(KNN). We can measure the efficiency of the classifier SVM based on the different kernel functions are linear and radial basis function(RBF). The classification of the documents based on the selection of the different kernel function will defines the feature space of the training dataset of the documents. And then KNN classifier is used to classify the documents based on the different parameters are electing the number of features, number of neighbors and then distance metric. Based on the number of features the efficiency of the KNN algorithm is increases. The resulting accuracy of both the algorithm provides the better accuracy. The SVM linear kernel classifier provides the better classification accuracy than the K nearest neighbor classifier.

Table 1: Accuracy(%) of text categorization.

Classifier	Accuracy	Precision	Recall	F-Measure
SVM Linear	88.49	90.11	88.49	88.58
SVM RBF	85.84	88.59	85.84	86.01
KNN	85.84	88.59	85.84	86.01

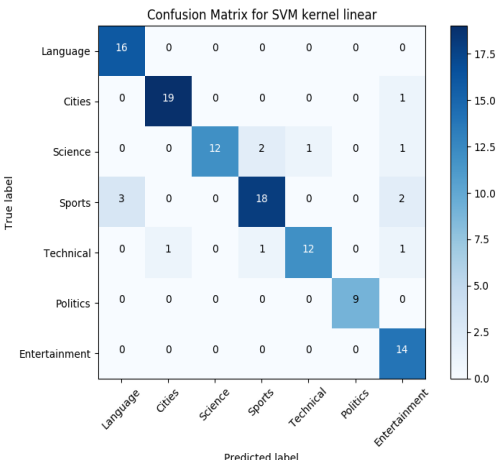

Figure 3: Confusion matrix for SVM linear kernel.

Figure 3 shows the confusion matrix for SVM linear kernel classifier. The x axis shows the predicted label and Y axis shows the True label. The predicted and true labels are Language, Cities, Science, Sports, Technical, Politics, and Entertainment. The correct classified labels values are 16, 19, 12, 18, 12, 9 and 14. Other than diagonal elements are misclassified labels.

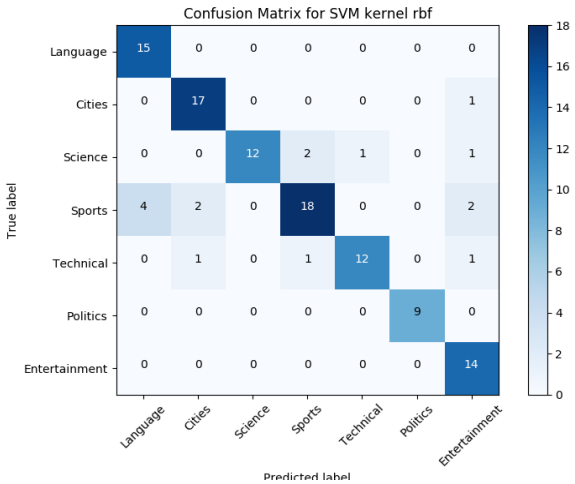

Figure 4: Confusion matrix for SVM rbf kernel.

Figure 4 shows the confusion matrix for SVM rbf kernel classifier. The x axis shows the predicted label and Y axis shows the True label. The predicted and true labels are Language, Cities, Science, Sports, Technical, Politics, and Entertainment. The correct classified labels values are 15, 17, 12, 18, 12, 9 and 14. Other than diagonal elements are misclassified labels.

```
[[13  0  0  0  0  0  0]
 [ 0 19  0  0  0  0  1]
 [ 1  1 12  2  2  0  1]
 [ 4  0  0 18  0  0  1]
 [ 0  0  0  1 11  0  1]
 [ 1  0  0  0  0  9  0]
 [ 0  0  0  0  0  0 15]]
('Accuracy:', 0.8584070796460177)
('Precision:', 0.88596221218687354)
('Recall:', 0.8584070796460177)
('F1:', 0.86018285340079614)
```

Figure 5: Confusion matrix and results for KNN classifier

Figure 5 shows the confusion matrix for K Nearest Neighbor classifier. The x axis shows the predicted label and Y axis shows the True label. The predicted and true labels are Language, Cities, Science, Sports, Technical, Politics, and Entertainment. The correct classified labels values are 13,19,12,18,11,9 and 15. Other than diagonal elements are misclassified labels.

CONCLUSION

Text categorization is one of the important applications in the field of the text classification. Here an automated text categorization approach to classify Wikipedia corpora is proposed. In this work, we are used two supervised machine learning algorithms for the classification of Wikipedia articles. The raw Wikipedia articles are initially pre-processed using python packages. Then the classifier is applied on these pre-processed corpora to classify them into different groups based on the pre-defined labels. Non linear, kernel based Radial basis function (RBF) and linear kernel of Support Vector Machine is used to obtain high efficiency during text categorization. Linear kernel of Support Vector Machine than the K nearest neighbor. Text categorization can be applied in the field of Health, banking, News etc. This experimented is conducted on an open source Wikipedia articles consisting of 564 files. 80% of the Wikipedia articles are dedicated to training and 20% of the articles are dedicated to testing. The experimental results showed that accuracy of the proposed system for Support Vector Machine and K Nearest Neighbor classifiers is 88% and 85% respectively

In future work, here we have listed some of the features which can be implemented in the future work.

- We can use machine learning algorithms like neural networks, decision tree to classify large volumes of unstructured data.
- We can merge supervised and un-supervised learning to organize text documents.
- We can experiment text categorization on large number of structured data sets in the future.

ACKNOWLEDGEMENT

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and the organization. I would like to extend my sincere thanks to all of them. I am grateful to my institution, **Ramaiah Institute of Technology** with its ideals and inspirations for having provided us with the facilities, which has made this, project a success.

REFERENCES

[1] W. Zhang, T. Yoshida, and X. Tang, "Text classification based on multi-word with support vector machine," Knowledge-Based Syst., vol. 21, no. 8, pp. 879-886, 2008.

[2] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, 'Word Cloud Explorer: Text Analytics Based on Word Clouds', 2014 47th Hawaii International Conference on System Sciences, Jan. 2014.

[3] S. Aurangabadkar and M. A. Potey, 'Support Vector Machine based classification system for classification of sport articles',2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Jan. 2014.

[4] L. H. Lee, R. Rajkumar, and D. Isa, "Automatic folder allocation system using Bayesian-support vector machines hybrid classification approach," Appl. Intell., vol. 36, no. 2, pp. 295-307, 2012.

[5] Vishwanath Bijalwan, Vinay Kumar, Pinki Kumari,

Jordan Pascual "KNNbased Machine Learning Approach for Text and Document Mining", International journal of database theory and application, Vol.7, No.1,pp.61-70, 2014.

[6] Pennacchiotti, Marco, and Ana-Maria Popescu. "A Machine Learning Approach to Twitter User Classification." ICWSM 11.1 (2011).

[7] Mita K. Dalal and Mukesh A. Zaveri, Automatic Text Classification: A Technical Review, Volume 28– No.2, and August 2011.

[8] A. Aizawa: An information-theoretic perspective of tf-idf measures, Information Processing and Management: an International Journal archive, Vol. 39, Issue 1, 2003, pp. 45-65.

[9] Ikonomakis, M., S. Kotsiantis, and V. Tampakas. "Text classification using machine learning techniques." WSEAS transactions on computers 4.8 (2005).

[10] Ioannis Antonellis, Christos Bouras, Vassilis Poulopoulos and Anastasios Zouzias, "Scalability of Text Classification", WEBIST (Web Interfaces andApplications), 2006.

[11] I. Sandu Popa, K. Zeitouni, G. Gardarin, D. Nakache, E. Metais, "Text Categorization for Multi-label Documents and Many Categories," Twentieth IEEE International Symposium on Computer-Based Medical Systems (CBMS'07), 2007.

[12] F. Sebastiani, Text Categorization, 2002.

[13] Wang, Y., and Wang X.J., " A New Approach to feature selection in Text Classification", Proceedings of 4th International Conference on Machine Learning and Cybernetics, IEEE- 2005.

[14] A.N. Nihthiya Althaf, N.Priya, Kalaivaazhi Vijayaraghavan, "Development of Novel Machine Learning approach for Document Classification", International Journal of Computer Trends and Technology (IJCTT) , 2017.

[15] Berry Michael W., Automatic Discovery of Similar Words, in "Survey of Text Mining: Clustering, Classification and Retrieval", Springer Verlag, New York, LLC, 2004.

[16] Vishal gupta and Gurpreet S. Lehal , "A survey of text mining techniques and applications", journal of emerging technologies in web intelligence, 2009.

[17] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin,"A: Practical Guide to Support Vector Classification.

[18] Bernhard Scholkopf and Alexander J. Smola ,"Learning with Kernels, Support Vector Machine, Regularization, Optimization and Beyond." The MIT press.

[19] Wang, Y., and Wang X.J., " A New Approach to feature selection in Text Classification", Proceedings of 4th International Conference on Machine Learning and Cybernetics, IEEE- 2005, Vol.6, pp. 3814-3819, 2005.

[20] Montanes,E., Ferandez, J., Diaz, I., Combarro, E.F and Ranilla, J., " Measures of Rule Quality for Feature Selection in Text Categorization", 5th international Symposium on Intelligent data analysis , Germeny-2003, Springer-Verlag 2003, Vol2810, pp.589598, 2003.