International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Natural Language Processing and Machine Learning Classifier used for Detecting the Author of the Sentence

M.Sheshikala, D. Kothandaraman , R.Vijaya Prakash, G.Roopa

Abstract*: Detecting the author of the sentence in a collective document can be done by choosing a suitable set of features and implementing using Natural Language Processing in Machine Learning. Training our machine is the basic idea to identify the author name of a specific sentence. This can be done by using 8 different NLP steps like applying stemming algorithm, finding stop-list words, preprocessing the data, and then applying it to a machine learning classifier-Support vector machine (SVM) which classify the dataset into a number of classes specifying the author of the sentence and defines the name of author for each and every sentence with an accuracy of 82%.This paper helps the readers who are interested in knowing the names of the authors who have written some specific words.

***Index Terms*: Natural Language Processing, Stemming, Stop words, Classifier.**

I. INTRODUCTION

A number of documents are written by 'n' number of authors, when reading a particular book or document, we may be eagerly interested in knowing the name of the author, who has written those sentences. According to the survey of google after counting, there are almost 210 million books around the world, among this an average women reads 14 books and man reads 9 books over a year. Generally when we try to search a book we get details of the author who has written the book, but we don't get the names of the author who has written the sentences in it. Few sentences in the reading book may be quite interesting, and wanted to know who has written that sentence, because every sentence written in the book may not be written by the same author of the book[1][2]. So, to know this we need to classify the reading document with each sentence written by an author. This can be done by Natural Language processing by inculcating the concepts of machine learning classifier.

Creator distinguishing proof isn't an examination territory that developed out of the expanded utilization of web. It was Utilized for figuring out which writer composed a part or section of a book, the good book being the most well known model. Creator recognizable proof research utilizes the structure of the content and the words that are utilized.

A subdivision of this is stylometric examine in which etymological qualities are utilized to recognize the Creator of content.

II. LITERATURE REVIEW

Marcia Fissette [12] has identified the author of the sentence with formal concept analysis using single words, but it is limited to a small text, the proposed work can be extended to long texts also.

Corney et al. [13] has listed 4 authors , with 253 messages , 0-964 (avg. 92) and identified that most successful words are character n-grams and function words, the author has conducted experiment with that 20 text samples of 100 words on 3 Ph.D. thesis considering up to 184 features, from 5 Different categories: 'character-based', 'function word frequency' , 'word-based', 'document-based', and 'word length frequency distribution'.

McCombe [14] , performed the tests using unigrams for identifying the sentence, but no method was used in classification based on word gram, so bigram placed a contradictory for future work.

Hirst and Feiguina [15] used bigrams for identification of the author sentence but the chance of properly classifying correctly without any features is done with 50 percentages.

In the past work as the number of messages increases per author the performance of classification increases and when the number of authors increases the performance of the classification decreases.

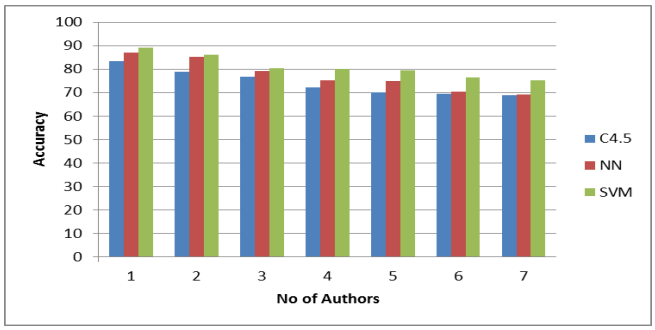

Figure 1: Performance of clasification decreases as the number of Authors increases.

Manuscript published on November 30, 2019.
* Correspondence Author
Dr.M.Sheshikala, CSE, S R Engineering College, Warangal, India.
Dr.D. Kothandaraman, CSE, S R Engineering College, Warangal, India
Dr.R.Vijaya Prakash, CSE, S R Engineering College, Warangal, India
G.Roopa, CSE, S R Engineering College, Warangal, India.

III. NATURAL LANGUAGE PROCESSING

NLP is the capacity of a PC program to comprehend human language as it is spoken. NLP is a segment of Artificial Intelligence.

NLP basically works using syntax and semantic techniques, where syntax is the words are arranged in a sentence and syntax includes parsing, word segmentation, morphological segmentation and stemming. Generally NLP applies algorithms to understand the meaning and structure of the sentences [3]. So using NLP we apply different steps and classify the data set, having with the information who has written the sentence. Before we start NLP, we need to pre-process the data.

A. Machine Learning:

It is the evident from the name, that machine learning is similar to humans [7]; it is a field of concentrate that enables the PC to learn without being expressly customized. For example, researchers or programmers are preparing the models to train the machines for detecting brain tumor just by looking at the slide. So here we prepare a model for applying NLP techniques first and later the input is to be given to the system, to train the model using machine learning classifier.

B. Pre-processing the Data:

Pre-processing data, which converts raw data into useful data [6] [4]. We have many techniques for data pre-processing:

1. Ignore the tuples
2. Fill the missing values
3. Binning Method
4. Regression
5. Clustering

When it is applied to Machine learning data needs to be further pre-processed by using the following techniques:

1. Rescale Data: Rescaling the attributes to have the same scale.
2. Binarize Data: Values can set to equal or less than 0 are marked 0 and all of those above 0 are marked 1.
3. Standardize Data: Data Can be standardize using scikit-learn with the Standard Scaler class

Various stages of NLP data cleaning are:

1. Finding whether unnecessary data is present, for example noisy such as symbols, specials characters etc.
2. Reducing the words to its root form using the techniques such as stemming or lemmatization.
3. Erasing the stop words such as an, the, is, for, then, etc.

Stemming:

Stemming is the process for removing the redundancy by reducing a word to its root form. For example, if eat, eaten and eating are present in a same sentence, then they are reduced to eat and counted as 3, not making each word as unique.

Stop Words:

These are the words which occur frequently in the document and are not necessary when comparing a document. For example, words like a, an, the, is, were and etc., are called as stop words.

C. Data Set

The dataset is based on English language literature by 10 famous authors. The train and the test data consists of short samples of text, where each sample consists of a set of 10 sentences. These sentences are irrespective of the number of words which constitutes the X data and the corresponding Y data, the author.

The training data and test data comprise of 18,977 and of 6,326 samples each. This is a dataset which has been collected over some time to gather works of the best authors from many generations.

Features Sample containing 10 sentences of English language text. Author of the corresponding text/sample (10 classes).

The implementation of algorithm- 1 is used for importing packages and this concept is done using python.

Algorithm 1:

Step 1: Panda, re and nltk packages need to be to be imported

Step 2: From different sub packages like extrac.text import countvectorizer.

Step 3: Import Confustion matrix

The required packages are downloaded and applied to the data set. The following is the code used to move through every perception in the dataset, evacuating exceptional characters, performing stemming and expelling stop words [1][10].

Algorithm-2: Actual procedure for classifying the data set.

1. the following link downloads the required stop words.
   nltk.download('stopwords')
2. corpus = [] # This is used to store scrub data.
3. #Initializing object for stemming
   ps = PorterStemmer()
   for i in range(len(df)): # for every study in df we see removal of special characters
   text34=re.sub12('[^a-zA-Z]','df['text'][i]). lower() . split()
   # removal of stemming and stop words.
4. Cleaned words are formed from a sentence
   text = ' '.join(text)
5. To the empty list the cleaned word are addded
   corpus.append(text)

NLTK libraby helps to come with a collection of stop words, which are used to clean the data set.

Stemming is performed by using the method nltk.stem.porter. After each observation we can see the removal of special characters in the data set, for this we use Porter Stemmer method.

0	str	1	pronounc expert rare varieti consider valu see handsom open flat box s ...
1	str	1	partner sail along front though notic noth medic student realli danc h ...
2	str	1	cushion good deal higher ball ball fashion alway stop cushion accompli ...
3	str	1	god grant presenc may rather see vile behav may benefit dinner fell as ...
4	str	1	grass glow bright fragrant flower bird fli flock air perch fearlessli ...
5	str	1	wake ask cutlet bottl chateau yquem grape paper ink bill one notic any ...
6	str	1	ask stolen someth said mitya wink slyli stolen inquir pyotr ilyitch cu ...
7	str	1	command neither less valu command man weak transgress command whatev c ...
8	str	1	particularli unkempt person look fix idea eye would get anyth interest ...
9	str	1	catesbi ten captain sir richard would march forthright london sir clai ...

Figure 2: Cleaned data after applying the corpus.

C. Generating the Count Vectors:

```
cv12 = CountVectorizer(features_mx = 80)
X1 = cv.fit12_transform(corpus).toarray()
y1 = df.iloc[:, 1].values
```

From the above code we can see Bag-of Words, and a matrix of vectors is created using the method called sklearn.feature_extraction.text in a sentence. 80 unique words are selected with the parameter max=features=80. Count Vector is created after transforming cleaned data which is used as separate variable for the test classifier.

We can see the following results in coming steps:

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	2	0
2	0	1	0	0	0	1
3	1	0	0	1	1	0
4	1	0	0	0	1	0

Figure 2: Observation of a row and column to represent 80 selected words.

The vector of X1 is represented as

```
[[1 3 1 1 2 1 2 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 1 1 1
 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1
1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1
1 1 1 1 1
 1 1 1 1 1 1 1 1 1]]
```

The vector of y1 is represented as

```
[[1 3 1 1 2 1 2 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 1 1 1
 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1
1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1
1 1 1 1 1
 1 1 1 1 1 1 1 1 1]]
```

The sample data printed from the dataset consisting the author sentences as below:

['ever cherish I'll feel miss maim forgav brave kind pretti venturesom decis weather frown]

D. Evaluating the model:

In the wake of foreseeing for the approval set, we have to check what number of the expectations are really right [1] [8].To do this, we will utilize the perplexity lattice. Utilizing the perplexity framework we will think about the anticipated qualities in y1_pred and the real qualities in y1_val. The precision from a disarray lattice can be determined by summing up the inclining components and jumping it by the all-out aggregate of components in the matrix [1] [9] [11]. We characterize a strategy as demonstrated as follow

Algorithm 3: In this algorithm we evaluate the confusion matrix.

Step1: Function defined for Confusion matrix .
Step2: Tracing of the matrix is done
Step 3: Adding up of all elements.
Step 4: returning the diagonal elements and printing the accuracy of the classification.

```
def accuracy12(Con_mat):
dia_sum12 = con_matr.trace()
 Sumofele = con_mat.sum()
 return diagonal_sum / sumofallele
cm 12= con_mat(y_val12, y_pred12)
print("Accuracy12 : ", accuracy(cm))
```

It is observed that as the accuracy of the Machine Learning Classifier increased when compared to the previous work. but as the number of authors increases the performance of classification decreases.

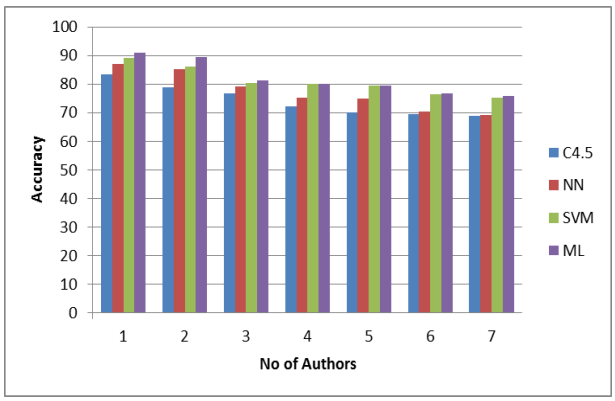

Figure 3: As the number of Authors increases the performance of the machine learning classifier decreases but with an improvement over other classifiers.

IV. CONCLUSION

Detecting the author of the sentence in a collective document is done by choosing a suitable set of features and implementing using Natural Language Processing in Machine Learning. Accuracy: 0.4461271882975549, from the observation 84% of prediction were true. Further this observation can be improved to 90% by taking large data set with more number of training and test samples.

REFERENCES

1. https://searchbusinessanalytics.techtarget.com/definition/natural-language-processing-NLP
2. R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in
3. Advances in neural information processing systems, 2015, pp. 3294–3302.
4. https://www.analyticsindiamag.com/how-to-solve-your-first-ever-nlp-classification-challenge/
5. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," DTIC Document, Tech. Rep., 1985.
6. D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.

7. S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuousspace," arXiv preprint arXiv:1511.06349, 2015.
8. Y. Zhang, Z. Gan, and L. Carin, "Generating text via adversarial training," in NIPS workshop on Adversarial Training, 2016.
9. Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, "Controllable text generation," arXiv preprintarXiv:1703.00955, 2017.
10. L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: sequence generative adversarial nets with policy gradient," in Thirty-FirstAAAI Conference on Artificial Intelligence, 2017.
11. Praveen, Ali, Sampath, Vijaya Prakash, Research on Multi-Agent Experiment in Clustering, International Journal of Recent Technology and Engineering". 2019.
12. Marcia Fissette, dr. F.A. Grootjen, "Author identification in short texts", DTIC Document, 2010.
13. M. Corney, A. Anderson, G. Mohay, and O. de Vel. Identifying the authors of suspect mail. Communications of the ACM, 2001.
14. Niamh McCombe. Methods of author identification. Master's thesis, Trinity College, Dublin Ireland, 2002.
15. G. Hirst and Ol'ga Feiguina. Bigrams of syntactic labels for authorship discrimination of short texts.Literary and Linguistic Computing, 22(4), 2007.

AUTHORS PROFILE

Dr.M.Sheshikala, completed her Ph.D from K L Educational Foundation in Computer Science and Engineering, Andhra Pradesh in March 2018. She is an Associate professor in the department of CSE at S R Engineering College.Her research interests are related to Data Mining, Machine Learning. She had published 28 publications in various national and international journals, conferences and proceedings. Her total teaching experience is 14 years.

D. Kothandaraman received his B.E. CSE from Dr. Pauls Engineering College (Anna University), M.Tech., in CSE-IS from PEC, Ph.D. in CSE from, Anna University(Govt. of Tamil Nadu), College of Engineering, Guindy. To his credit, he has 8 years of teaching and research experience. His area of research interest is computer networks, Wireless Sensor Networks (WSN), Mobile Ad-hoc Networks (MANETs) and Internet of Things (IoT). He has published various papers in International Journals and in conferences. Currently, he is working as Associate Professor in CSE Department at S R Engineering College, Warangal

Dr. R. Vijaya Prakash is working as a Professor in the Department of Computer Science and Engineering, S R Engineering College, Warangal. He completed his Ph.D. from Kakatiya University, Warangal in 2014. His area of interest is Data Mining, Artificial Intelligence, and Assistive Technology. He published various papers in International Conferences and Journals.

G.Roopa, presently working as Assistant professor in the Department of Computer Science and Engineering, S R Engineering College, Warangal. He has total 10 years of teaching experience and her research interest are Network Security, Cloud Computing.