IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 3, March 2022

DOI: 10.17148/IJARCCE.2022.11322

News Article Sentimental Analysis Using Modified Na¨ıve Bayes' Algorithm

Prof. Laxmi Pawar[1], Mr. Ankush Bhalerao[2], Prof. Sachin Jagdale[3]

Assistant Professor, Computer, Pune Institute of Computer Technology,Pune, India[1]

Student, Computer, Pune Institute of Computer Technology, Pune, India[2]

Assistant Professor, IT,Vishwakarma Institute of Information Technology, Pune, India[3]

Abstract: Opinions of the public and the sentiments originating thereby play a pivotal role in social procedures. Sentiment analysis deals with the resolution of the tone or polarity of the text- how positive or negative it is. When applied to news reports,it provides a wide range of applications.

This study analyses news reports in real-time from reliable sources using a slightly modified Na¨ıve Bayes' Algorithm. An article is fetched and then pre- processed to get rid of noisy words like English articles. After tokenization, the probability of each word being either positive or negative is determined. This is achieved by training a model using a dataset of brief news headlines, with their sentiment values labelled. The overall probability is summed using the well-knownBayes' theorem, which gives the name to the algorithm.A slight modification is proposed to this algorithm by calculating sentiment value for the field 'engineering,' which separates or calculates how a particularreport is related to engineering. Based on the relevance to engineering (defined herewith using the dataset), a system is developed that prompts the head of an organization or any competent authorities with the report through an email

Keywords: text analysis, natural language processing (NLP), machine learning, text polarity, opinion mining, Na¨ıve Bayes'.

I. INTRODUCTION

Opinions are one of the highly influential factors for human social behaviour. Quite often, one sees the world through the eyes of others. An individual re- lies on the words of family and friends for decision-making. When scaled to the gargantuan world, people express scores of thoughts on various platforms. The internet quite boldly hoisted the jack for the freedom of speech, but in its wake, numerous outlooks and ideas have been accumulating by opinions, articles, re- views, tweets, microblogs, blogs, forums, and similar such spaces. It takes trivial efforts for a person to express estimations through comments on social media. And most significantly, the mirror of society, quite often described as the fourth pillarof democracy- the media also reports events through online news portals. These thoughts or reports can be subjected to the sentimental analysis.

Sentiment analysis is a text mining challenge that deals with determining the opinion expressed by the author of a text. The most common approach to text sentiment analysis is detecting the value of features (words in the text)- positive or negative.

Machine learning is employed for tone detection for the plethora of data avail- able on the internet. This study proposes a probabilistic Na¨ıve Bayes' algorithmto determine the tone or the sentiment of the text. The sentiment can take oneof these values: positive (1), negative(-1) or neutral (0). This machine-learning algorithm is applied to online news articles using natural language processing tech- niques (NLP).

The Na¨ıve Bayes' approach is applied to determine the probabilities of classes assigned to the text by using joint probabilities of words and classes. A class is a sentiment value, so we have two classes. Firstly, a model is trained to distinguishthe text and prepare sentiment dictionaries for all the classified text data. These dictionaries store the value of the probability of that feature(word) belonging tothat class. The test vector, in this case, is a new vectorized report. The proba-bility of whether that vector belongs to these classes is calculated separately for every class. All these probabilities make use of the mathematical Bayes' theoremfor calculation, hence the name. A relativistic comparison of probabilities for both classes gives the sentiment of the test data. The class with which the vector yieldsa higher probability value is declared to be its sentiment.

A modification in the algorithm suggests a new class named 'engineering' par-allel to the existent classes: 'positive' and ‘negative.’ The probability of whether the test vector belongs to this class is calculated. If this figure passes a pre-decided threshold value, the head of an organization or any concerned authorities is given a prompt with this system using an email. The news articles for testing are selectedfrom highly reliable news portals that directly report any event straight-forward. It means the language should essentially make minimal use of sarcasm or puns orany hidden implications. The machine is taught to process the text. This articleis then processed using various techniques to make it suitable for analysis. This system uses Python and its libraries like nltk (natural language toolkit) and pan-das for providing some crucial functions.

The dataset used is acquired from Kaggle. It is compiled online and consistsof short headlines labelled positive or negative.

This model finds a hindrance when it comes across sentences that have eu- phemisms or words with hidden implications. Sarcasm would be treated unchecked and taken literally since the model is probabilistic in nature.

So, this study overall consists of the following steps:

1. Data collection and pre-processing
2. Data mining using Na¨ıve Bayes’ algorithm
3. Result and its report using the developed system

II.PROBLEM DEFINITION

To propose an algorithm to classify the news reports fetched from online newsportal into one of the sentiments: positive or negative. Further, check the relativity of the news article with the field engineering and report the same to the authoritiesif closely related.

A.Scope

The algorithm calculates probability of class belonging for the test data. The sen- timent with higher probability becomes the overall sentiment of the news report. Further, the dataset for engineering field can be enriched for specified news article detection. Large dataset can improve accuracy for the sentiment detection.

The degree of polarization of the text can also be determined. This can be done by introducing a scale for each class. The extent of positivity or negativity could possibly be judged on the basis of difference between the class probabilities. Higher the difference, higher the degree of polarization. For the engineeringsentiment(class) the approach remains as mentioned with the threshold value.

III.ALGORITHMS USED FOR SENTIMENT ANALYSIS

A. Naïve Bayes’ Algorithm

It is an easy-to-build algorithm and proves efficient for large datasets too. It classifies the text based on probabilities. Na¨ıve implies the independence of oc- currence of a feature from occurrences of the other features, whereas the word Bayes’ suggests the use of the Bayes’ theorem. Bayes’ theorem determines the probability of a hypothesis with prior knowledge. It depends on conditional prob- ability. Initially, the dataset is arranged into frequency tables for each class. The classes are: positive and negative. For this, the dataset is preprocessed. Then the sentiment dictionaries are obtained for the given dataset that carries likelihoodprobabilities for the features. Finally, the Bayes’ theorem is used to calculate the posterior probability for the test data- news report in this case. The class witha higher Bayes’ probability becomes the sentiment of the article. With a slightmodification, an additional class is introduced called ‘engineering’ for checking the relevance of the article to the field of engineering. The other steps take place parallelly and the probability obtained using the Bayes’ theorem is tested againsta threshold value that determines the relevance. Along with sentiment analysis,the Na¨ıve Bayes’ classifier is used for real-time prediction, multiclass prediction, personalized recommendation systems.

B.Porters Stemmer Algorithm

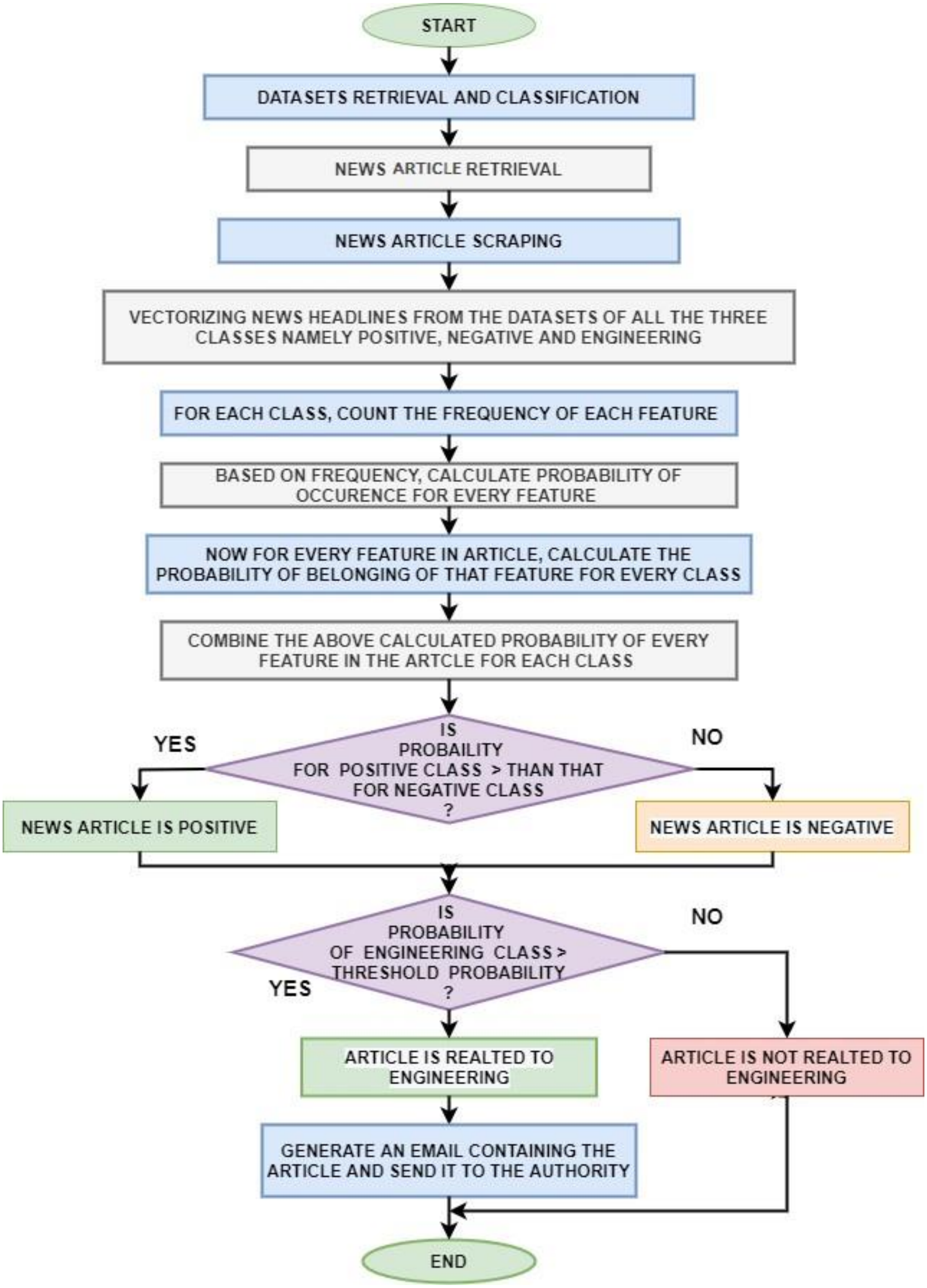
The process of reducing a word to its root(lemma) is known as stemming. Portersalgorithm can be used for this purpose. It studies the features based on vowels and consonants to find a pattern and strip away the suffix or prefix. It is a significant aspect of normalization while setting up Information Retrieval (IR) systems.

DOI: 10.17148/IJARCCE.2022.11322

IV.METHODOLOGY

A.Workflow

Figure 1: Sentimental Analysis framework: Workflow


START
DATASETS RETRIEVAL AND CLASSIFICATION
NEWS ARTICLE RETRIEVAL
NEWS ARTICLE SCRAPING
VECTORIZING NEWS HEADLINES FROM THE DATASETS OF ALL THE THREE CLASSES NAMELY POSITIVE, NEGATIVE AND ENGINEERING
FOR EACH CLASS, COUNT THE FREQUENCY OF EACH FEATURE
BASED ON FREQUENCY, CALCULATE PROBABILITY OF OCCURENCE FOR EVERY FEATURE
NOW FOR EVERY FEATURE IN ARTICLE, CALCULATE THE PROBABILITY OF BELONGING OF THAT FEATURE FOR EVERY CLASS
COMBINE THE ABOVE CALCULATED PROBABILITY OF EVERY FEATURE IN THE ARTCLE FOR EACH CLASS
IS PROBAILITY FOR POSITIVE CLASS > THAN THAT FOR NEGATIVE CLASS ?
YES
NO
NEWS ARTICLE IS POSITIVE
NEWS ARTICLE IS NEGATIVE
IS PROBABILITY OF ENGINEERING CLASS > THRESHOLD PROBABILITY ?
NO
YES
ARTICLE IS REALTED TO ENGINEERING
ARTICLE IS NOT REALTED TO ENGINEERING
GENERATE AN EMAIL CONTAINING THE ARTICLE AND SEND IT TO THE AUTHORITY
END

A model is trained using datasets that have headlines labelled withsentiment values ‘positive’ or ‘negative’. These values can be referred to as classes. These headlines are segregated, into corresponding lists, after which they are processed to form sentence vectors in which every word in a headlineis tested against every word in the complete its class. This commences the

DOI: 10.17148/IJARCCE.2022.11322

algorithm. Dictionary of counts of occurrences is prepared for each class. And then the most crucial step of creation of sentiment dictionary takes place in which probability of each word belonging to its class is calculated and stored against that particular word. This essentially depends upon the count of occur-rences of that word. This implies the more the word is repeated throughout itsclass, the higher is its probability as compared to the other words in the same class.

Now the above steps are identically repeated for the test dataset, in thiscase, the news report. Its features are vectorized, counted and dictionary of counts of occurrences is prepared. For each class, a dictionary of probability for each word belonging to that class is prepared by using the values from the already existent dictionary of counts of occurrences. There might be some wordsin the article that could be completely foreign to both our classes to producea zero probability. Laplace smoothing has been added to the original Na¨ıve Bayes approach to avoid the problem of zero probability that might nullify the entire probabilistic equation. The final probability is calculated taking into considerations all words (including repetitions) for each class dictionary. These probabilities could be extremely small but can still be relatively compared among classes. The class with higher probability is deemed to be the overall sentiment.

In this algorithm, an important modification is made by creating another class having sentiment value ‘engineering’. This is done to check whether the article is related to the field engineering and the extent of relation. For this, two words: ‘engineering’ and ‘college’ are searched for initially. Only if the article contains these two, the further processes take place.

Corresponding training dataset for engineering related articles has been pre-pared and would be processed in similar manner, but parallelly along with the other classes. In the final step we get Na¨ıve Bayes’ probability for each class. Forthis newly introduced class, there is no opposite class existing. In simple words, any relativistic comparison here is meaningless. So, a threshold probability is de- cided. If the test dataset gives probability higher than this probability, the textis considered to be related to engineering. an email consisting of this article is generated.

B. Mathematical Model

This algorithm is based on Bayes’ theorem stated as follows:

$$P(A|\ B) = \frac{P(B|A) * P(A)}{P(B)} \qquad (1)$$

Let P be the set of positive articles, N be the set of negative articles, andE is the set of articles related to engineering. Let U be the universal set of all the reports. Let A be the test article.

We represent the probability that A belongs to P as:

$$P(P|A)$$

, the probability that A belongs to N as:

$$P(N|A)$$

And the probability that A belongs to E as:

$$P(E|A)$$

As per Bayes’ theorem, these can be calculated as: (2)

$$P(P|\ A) = \frac{P(A|P) * P(P)}{(P(A)}$$

And,

$$P(N\ |A) = \frac{P(A|N) * P(N)}{P(A)} \qquad (3)$$

And,

$$P(E|\ A) = \frac{P(A|E) * P(E)}{P(A)} \qquad (4)$$

An important point to note is that we are doing the relativistic study be-tween equations 1 and 2 and a similar threshold

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁙ Vol. 11, Issue 3, March 2022

DOI: 10.17148/IJARCCE.2022.11322

comparison after finding resolving equation 3.
Therefore, the denominators, being the same in all the equations, can beneglected.
Now, find second terms in each of the above equations 1, 2, and 3.

$$P(P) = \frac{n(P)}{n(U)} \tag{5}$$

$$P(N) = \frac{n(N)}{n(U)} \tag{6}$$

$$P(E) = \frac{n(E)}{n(U)} \tag{7}$$

For the first terms in the numerator of the equations 1, 2 and 3,

$$P(A|P) = P(\frac{a_1}{P}) * P(\frac{a_2}{P}) * \ldots . * P(\frac{a_n}{P}) \tag{8}$$

Where a_n are the features of A. Because the reports do not appear as it is inthe dataset, but the words do.Similarly, the above values are obtained for N and E too.
Similarly, the first terms in the numerators are calculated by substituting the values for corresponding sets.
For every feature the probability that it belongs to any of the set is calculated as follows:

$$P(a_i|P) = \frac{n(a_i)\ in\ P}{n(P)} \tag{9}$$

Similarly,

$$P(a_i|N) = \frac{n(a_i)\ in\ N}{n(N)} \tag{10}$$

And,

$$P(a_i|E) = \frac{n(a_i)\ in\ E}{n(E)} \tag{11}$$

Thus the values for the equations 1, 2, and 3 are calculated and sentiment is given as

$$Sentiment = \begin{cases} POSITIVE, & if\ \ P(P|A) > P(N|A) \\ NEGATIVE, & if\ \ P(N|A) > P(P|A) \end{cases} \tag{12}$$

And for calculation of relevancy with engineering, comparison is done with engineering threshold probability i.e $P_{th}(E)$,

$$RELEVANCY = \begin{cases} Relevant, & if\ \ P(E|A) > P_{th}(E) \\ Not\ relevant, & if\ \ P(E|A) < P_{th}(E) \end{cases} \tag{13}$$

C.Key Terms

P(A/B) : probability of event A given event B is trueP(B/A) : probability of event B given event A is trueA: News Article

U: Universal set of all articlesP: Set of positive articles

N: Set of negative articles

E: Set of articles related to engineeringn(P): Number of articles in P

n(N): Number of articles in N n(E): Number of articles in E n(U): Total number of articles

a_i : *feature of an article* (*word*)

$P_{th}(E)$: *Engineering threshold probability.*

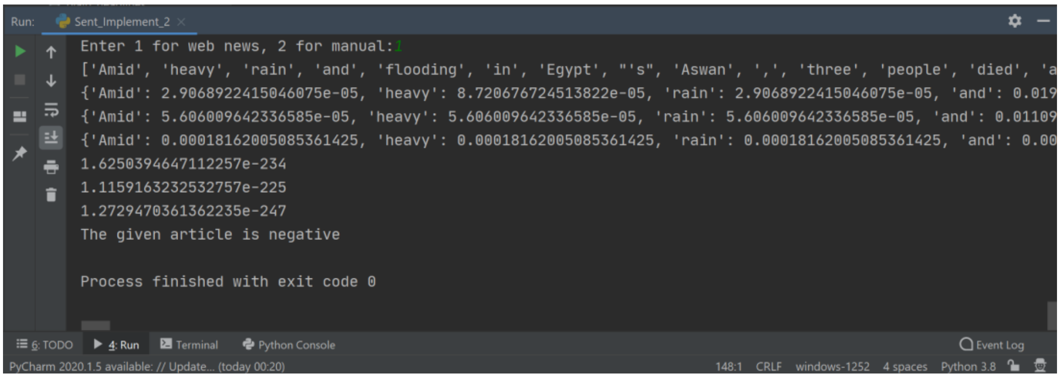
V.ALGORITHM

```
1.  Load the dataset in the data-frame training data1;
2.  pos docs1 = [] , neg docs1 = [], engg docs1 = [];
3.  for each sentiment, news in training dataset1 do:
if sentiment == ‘positive’: pos docs1.append(sentence);else if sentiment == ‘negative’:
neg docs1.append(sentence);

Else:
engg docs1.append(sentence);

end ifend for
4.  Vectorize the lists pos_docs1, neg_docs1 and engg_docs1 and load tothe data-frame pos vec1, neg
vec1 and engg vec1 respectively;
5.  Count the frequency of features(words) from vectors for each list and storethe feature and its occurrence
as key- value in dictionaries freq_pos1,freq neg1 and freq_engg1;
6.  for feature, frequency in freq pos1:
calculate the probability of occurrence for feature using formula (9); store the feature as key and probability as
value in sentiment dictionary
called ‘po’;
end for;
7.  for feature, frequency in freq neg1:
calculate the probability of occurrence for feature using formula (10); store the feature as key and probability as
value in sentiment dictionary
called ‘ne’;
end for;
8.  for feature, frequency in freq engg1:
calculate the probability of occurrence for feature using formula (11); store the feature as key and probability as
value in sentiment dictionary
called ‘eni’;
end for;
9.  Fetch the news article from the website as ‘new sentence1’;
```

DOI: 10.17148/IJARCCE.2022.11108

```
10.    Tokenize new sentence1;
11.    for each feature in new sentence1:
           Calculate probability of containership of feature to the freq pos1 asnew prob1 based on formula (8) and store in dictionary;
           Calculate probability of containership of feature to the freq neg1 asnew prob2 based on formula (8) and store in dictionary;
           Calculate probability of containership of feature to the freq engg1 asnew prob3 based on formula (8) and store in dictionary;
       end for;
12.    Calculate the final probability by taking products of all individual prob- ability calculated above and store as final prob pos1, final prob neg1, fi- nal prob engg1;
13.    If final prob pos1 > final prob neg1:Sentiment = positive;
       Else:
           Sentiment = negative;End if
14.    Print sentiment.
15.    If final prob engg1 > threshold prob:
           Print "The article is related to engineering";Generate an email containing the article; Send it to the specified user id;
       Else:
           Print "The article is not related to engineering";
       End if End algorithm
```

VI.RESULT ANALYSIS

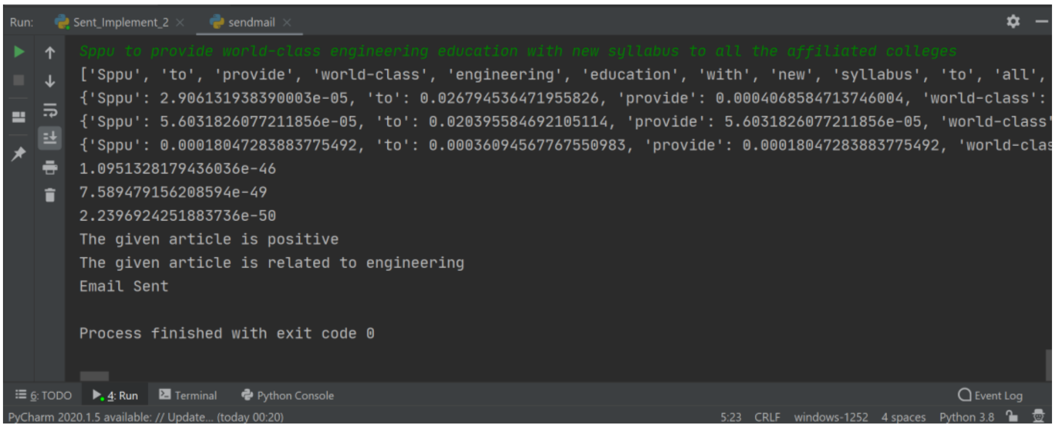


```
Enter 1 for web news, 2 for manual:1
['Amid', 'heavy', 'rain', 'and', 'flooding', 'in', 'Egypt', "'s", 'Aswan', ',', 'three', 'people', 'died', 'a
{'Amid': 2.9068922415046075e-05, 'heavy': 8.720676724513822e-05, 'rain': 2.9068922415046075e-05, 'and': 0.019
{'Amid': 5.606009642336585e-05, 'heavy': 5.606009642336585e-05, 'rain': 5.606009642336585e-05, 'and': 0.01109
{'Amid': 0.00018162005085361425, 'heavy': 0.00018162005085361425, 'rain': 0.00018162005085361425, 'and': 0.00
1.6250394647112257e-234
1.1159163232532757e-225
1.2729470361362235e-247
The given article is negative

Process finished with exit code 0
```

Run: Sent_Implement_2
6: TODO
4: Run
Terminal
Python Console
Event Log
PyCharm 2020.1.5 available: // Update... (today 00:20)
148:1 CRLF windows-1252 4 spaces Python 3.8

Figure 2: Real time classification of an article: Result

Number of articles have been fetched from the website inshorts.comand the first article is chosen for analysis. In the above image, it is clearly visible that the probability of the article belonging to the neg- ative class is higher than positive class. From inspection of the ini-tial words such as 'died' and 'flooding', the negativity of the news ar- ticle is evident. Therefore, it is rightly classified as a negative article.



```
Sppu to provide world-class engineering education with new syllabus to all the affiliated colleges
['Sppu', 'to', 'provide', 'world-class', 'engineering', 'education', 'with', 'new', 'syllabus', 'to', 'all',
{'Sppu': 2.906131938390003e-05, 'to': 0.026794536471955826, 'provide': 0.0004068584713746004, 'world-class':
{'Sppu': 5.6031826077211856e-05, 'to': 0.020395584692105114, 'provide': 5.6031826077211856e-05, 'world-class'
{'Sppu': 0.00018047283883775492, 'to': 0.00036094567767550983, 'provide': 0.00018047283883775492, 'world-clas
1.0951328179436036e-46
7.589479156208594e-49
2.2396924251883736e-50
The given article is positive
The given article is related to engineering
Email Sent

Process finished with exit code 0
```

Run: Sent_Implement_2
sendmail
6: TODO
4: Run
Terminal
Python Console
Event Log
PyCharm 2020.1.5 available: // Update... (today 00:20)
5:23 CRLF windows-1252 4 spaces Python 3.8

Figure 3: Checking relevancy with energy and generating email: Result

A line of an article saying ‘Sppu to provide world-class engineering education with new syllabus to all the affiliated colleges’ is visible. It is clearly related to engineering and from inspection it appears positive. The same resultis reflected by our model and hence an email is sent to a specified email id.

VII. CONCLUSION

News articles retrieved from the online portal are classified as positive or negative and checked for consistency in the class engineering. The Na¨ıve Bayes Classifier algorithm predicts the probability of which class, whether positive or negative, does the test data belongs. It is done based on the training datasets, hence enriching the dataset with more and more sample data increases the algo- rithm’s efficiency. The aspects that the model struggles to deal with are satire and parody. According to the observations, the proposed model is around 75 percent accurate, with the current pool of 3000 training news reports. The precision will enhance with the enrichment of the dataset.

The introduced modification detects the articles related to engineering or could be in the interest of an organization (here Pune Institute of Computer Technology), and authority is alerted in the form of an automatically generated email.

REFERENCES

[1] J. Kim, J. Seo, M. Lee and J. Seok, “Stock Price Prediction Through theSentimental Analysis of News Articles,” 2019 Eleventh International Confer- ence on Ubiquitous and Future Networks (ICUFN), 2019,*pp. 700-702, doi: 10.1109.*

[2] H. Al-Sarhan, M. Al-So’ud, M. Al-Smadi, M. Al-Ayyoub and Y. Jararweh, “Framework for affective news analysis of Arabic news: 2014 Gaza attacks case study,” 2016 7th International Conference on Information and Commu-nication Systems (ICICS), 2016, pp. 327-332, doi: 10.1109

[3] X. Wang and X. Luo, “Sentimental Space Based Analysis of User PersonalizedSentiments,” 2013 Ninth International Conference on Semantics, Knowledgeand Grids, 2013, pp. 151-156, doi: 10.1109

[4] K. Mizumoto, H. Yanagimoto and M. Yoshioka, “Sentiment Analysis of Stock Market News with Semi-supervised Learning,” 2012 IEEE/ACIS 11th Inter-national Conference on Computer and Information Science, 2012, pp. 325-328, doi: 10.1109/ICIS.2012.97.

[5] P. Khurana Batra, A. Saxena, Shruti and C. Goel, “Election Result Prediction Using Twitter Sentiments Analysis,” 2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC), 2020, pp. 182-185, doi:10.1109/PDGC50313.2020.9315789.

[6] S. Rana and A. Singh, “Comparative analysis of sentiment orientation using SVM and Naive Bayes techniques,” 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), 2016, pp. 106-111, doi: 10.1109

[7] F. Sağlam, H. Sever and B. Genç, “Developing Turkish sentiment lexicon for sentiment analysis using online news media,” 2016 IEEE/ACS 13th Interna- tional Conference of Computer Systems and Applications (AICCSA), 2016,pp. 1-5, doi: 10.1109/AICCSA.2016.7945670.

[8] Y. Gao, P. Su, H. Zhao, M. Qiu and M. Liu, “Research on Sentiment Dic- tionary Based on Sentiment Analysis in News Domain,” 2021 7th IEEE IntlConference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Con-ference on High Performance and Smart Computing, (HPSC) and IEEE IntlConference on Intelligent Data and Security (IDS), 2021, pp. 117-122, doi: 10.1109/BigDataSecurityHPSCIDS52275.2021.00031.

[9] V. Ikoro, M. Sharmina, K. Malik and R. Batista-Navarro, ”Analyzing Senti- ments Expressed on Twitter by UK Energy Company Consumers,” 2018 Fifth International Conference on Social Networks Analysis, Management and Se-curity (SNAMS), 2018, pp. 95-98, doi: 10.1109/SNAMS.2018.8554619.