(ISO 9001:2015 Certified), Accredited with ‘A’ Grade by NAAC)

🕿: 08258 - 281039 – 281263, Fax: 08258 – 281265

**Department of Computer Science and Engineering**

B.E. CSE Program Accredited by NBA, New Delhi from 1-7-2018 to 30-6-2021

**Course Code: 16CS704**

**Course Name: Machine Learning**

**News Category Classification using Naive Bayes Algorithm**

Semester: 7th Section: C

Submitted by:

Prem Stevenson Pereira 4NM16CS108

Rahul D Shetty 4NM16CS111

Submitted to:

**Mrs. Divya Jennifer D’Souza**

Department of Computer Science and Engineering

**Date of submission: October 14, 2019**

**Signature of Course Instructor**

Table of Contents

[Abstract 3](#_Toc21977257)

[Literature Survey 3](#_Toc21977258)

[Methodology 3](#_Toc21977259)

[Naïve Bayes Classifier 3](#_Toc21977260)

[Dataset 4](#_Toc21977261)

[Procedure 4](#_Toc21977262)

[Pre-processing: 4](#_Toc21977263)

[Feature Engineering 4](#_Toc21977264)

[Results 5](#_Toc21977265)

[Conclusion 6](#_Toc21977266)

[References 6](#_Toc21977267)

# Abstract

For the last few years, text mining has been gaining significant importance. Since Knowledge is now available to users through variety of sources i.e. electronic media, digital media, print media, and many more. Due to huge availability of text in numerous forms, a lot of unstructured data has been recorded by research experts and have found numerous ways in literature to convert this scattered text into defined structured volume, commonly known as text classification.

The major work is performed for news classification in the field of text mining. In this project we aim at classifying the news based on the headlines by using the Naïve-Bayes classifier.

# Literature Survey

The article written by Dr. Sebastian [1] describes how Naïve Bayes Algorithm can be utilized to classify a given email message as spam or ham. Bag of Word model is used which was constructed after pre-processing an email message through stop-word removal, stemming and lemmatizing. Natural Language Toolkit [2] is an open source program developed to perform various text processing tasks. This package defines methods for stop word removal, lemmatization and stemming. The article [3] is based on n-gram model for spam filtering where they found out that by using a n-gram of size 3 and 4 would produce better results when compared to the normal unigram model.

The web article [4] shows a step by step expressions and values for deciding if a sentence is sports or non-sports related. Here they implemented a count based approach and used this as prior for calculating the conditional probabilities. They also show how to make use of Laplace Smoothing to prevent probabilities to always be zero. Another interesting web article [5] which uses the concept of logarithms to prevent underflow while working with large denominators. On applying the logarithm for each term of the Bayes Theorem, the multiplication would be converted to addition and division would be a simple subtraction of the logarithms of each terms.

# Methodology

## Naïve Bayes Classifier

Naive Bayes classifiers are linear classifiers that are known for being simple yet very efficient. The probabilistic model of naive Bayes classifiers is based on Bayes’ theorem, and the adjective naive comes from the assumption that the features in a dataset are mutually independent.

In probability theory and statistics, Bayes’ theorem (alternatively Bayes’ law or Bayes’ rule) describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

Bayes’ theorem is stated mathematically as the following equation.

where A and B are events and P(B) ≠ 0.

* is the conditional probability : the likelihood of event A occurring given that B is true.
* is also a conditional probability : the likelihood of event B occurring given A is true.
* and are the probabilities of observing A and B independently of each other; this is known as marginal probability.

## Dataset

In the current project we make use of the BBC news dataset which consists of the head lines found in articles along with its corresponding tags. The dataset consists of 2000 samples of text which specifies the title and body of the news article and also the category. There are five categories of news articles which include *sport, business, technology, entertainment* and *politics*.

## Procedure

### Pre-processing:

The text got from the headlines cannot be used directly as it may contain some words and conjunctions that are useless for the classification. Thus pre-processing of the words is necessary to retain only the useful words. The following steps are involved

#### Removing contractions:

A contraction is a word or phrase that has been shortened by dropping one or more letters. In writing, an apostrophe is used to indicate the place of the missing letters.

#### Removing stop words:

In natural language processing, useless words (data), are referred to as stop words. Stop Words: A stop word is a commonly used word such as “the”, “a”, “an”, “in”.

#### Perform stemming:

Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words known as a lemma.

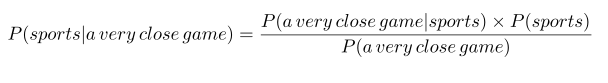
#### Perform lemmatization:

Lemmatization is the process of converting a word to its base form. The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form, whereas stemming just removes the last few characters, often leading to incorrect meanings and spelling errors.

### Feature Engineering

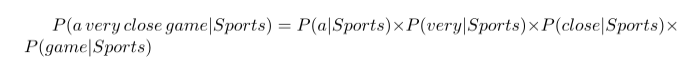
The important part is to find the features from the data to make machine learning algorithms works. In this case, we have text. We need to convert this text into numbers that we can do calculations on. We use word frequencies. That is treating every document as a set of the words it contains. Our features will be the counts of each of these words. Then, we need to convert the probability that we wish to calculate into a form that can be calculated using word frequencies.

In our case, the probability that we wish to calculate can be calculated as:

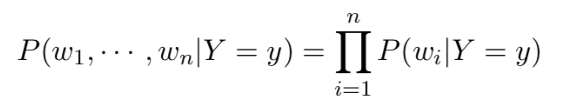


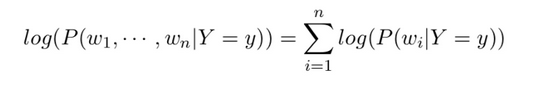
Similarly probabilities are calculated for every category.

Since there is an assumption that every word is independent of one another. Now we look at the individual words in the sentence rather than the whole sentence. Here we rewrite the probability we wish to calculate accordingly.



Prior to fitting the model and using machine learning algorithms for training, we need to think about how to best represent a text document as a feature vector. A commonly used model in Natural Language Processing is the so-called bag of words model.

If our document has more words then the probabilities will be very small which may lead to underflow problem. Hence we make use of logarithmic functions to maintain numerical stability.

Therefore the above probability equation can be rewritten as:

Since we are calculating the overall probability of the class by multiplying individual probabilities for each word, we would end up with an overall probability of 0 for the positive class. So we make use of smoothing algorithm such as Laplace smoothing.

#### Laplace Smoothing

We modify our conditional word probability by adding 1 to the numerator and modifying the denominator as such:

This can be simplified to

where |V| is our vocabulary size.

# Results

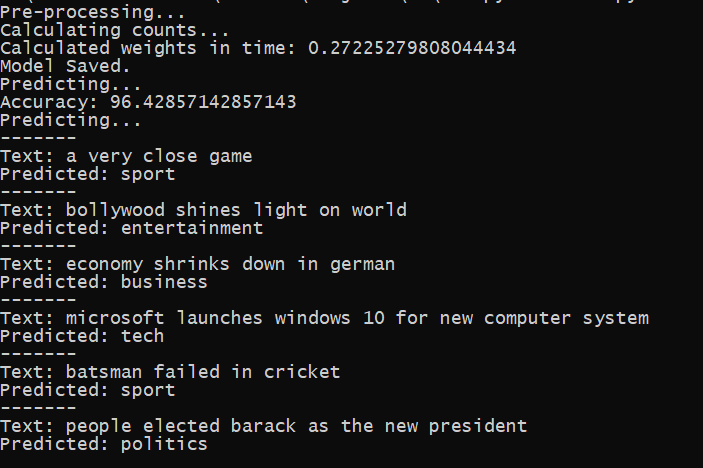
For the present experiment the Naïve Bayesian model is trained on the ‘BBC-text’ dataset which consists of head lines from various articles along with its category. The following figure shows the accuracy obtained while classifying the new input samples into our implementation of the Naïve Bayes algorithm.

Figure 1 Accuracy and the classification of some of the sample texts using our implementation.

The following figure shows the accuracy obtained by using the inbuilt Naïve Bayes classifier available in the Scikit-learn library.



Figure 2 Accuracy of classification using the algorithm built into sklearn

It is observed that the accuracy of the classification of our implementation of Naïve Bayes algorithm is equal to that of the one that is found in the machine learning libraries.

# Conclusion

News is important for a number of reasons within a society. Mainly to inform the public about events that are around them and may affect them. In the present scenario where the news is being delivered based on the readers interests, classification plays a major role. In this project we have successfully implemented the Naïve Bayes algorithm using for the classification of the news articles into distinct categories. This implementation can be further refined and used to classify any form of texts.\

# References

|  |  |
| --- | --- |
| [1] | D. S. Raschka, "Naive Bayes and Text Classification," 4 October 2014. [Online]. Available: https://sebastianraschka.com/Articles/2014\_naive\_bayes\_1.html. |
| [2] | E. Loper and S. Bird, "NLTK: The Natural Language Toolkit," 17 May 2002. [Online]. Available: http://www.nltk.org/. |
| [3] | I. KANARIS, K. KANARIS, I. HOUVARDAS and E. STAMATATOS , "WORDS VS. CHARACTER N-GRAMS FOR ANTI-SPAM FILTERING," *International Journal on Artificial Intelligence Tools ,* pp. 1-20, 2006. |
| [4] | B. Stecanella, "A practical explanation of a Naive Bayes classifier," 25 May 2017. [Online]. Available: https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/. |
| [5] | H. H. Nguyen, "Algorithms for Text Classification," 4 February 2019. [Online]. Available: https://towardsdatascience.com/algorithms-for-text-classification-part-1-naive-bayes-3ff1d116fdd8. |