**Client-Service-Rating: Applications of Machine Learning in Document Classification**

Surname: Manamela

Name: Tsholofelo  
Student Number: 201705648

Module Name: Advanced Machine Learning

*Abstract*

The aim of this study is to build a probabilistic model using the Naïve Bayes classifier to classify online course reviews without the use of packages. The data is extracted from website named Coursera. Naïve Bayes classification is based on the concept of conditional probability. It calculates features based on other existing attributes. This method originates from the estimation need of calculating multivariate probabilities using a training dadaset. The paper demonstrates the probability that a comment will fall under a certain class.

Keywords: Naïve Bayes, Probability, Text processing.

# **Introduction**

Reviews play a significant role in highlighting what people are feeling regardless of whether they are positive or negative. These reviews are critical opinions that are normally written in text, introducing the need for natural language processing systems[1]. Data is generated in large amounts every day leading to information overload. A lot of this data is retrieved using keywords through search engines and social media, hence the need for research such as semantic retrieval [8]. The issue is that computers do not have the ability to process text data by themselves, therefore pre-processing techniques must be applied before the development of any model [2]. The text data is converted to binary data. One of the most famous tools for processing text data is the Naïve Bayes classification method [3]. The naïve Bayes algorithm tries to build a classifier by using historic training data with class labels [4].

This paper describe how text data is pre-processed to form a training sample that can be used to build a classifier using the naïve Bayes method.

# **literature Review**

The naïve Bayes algorithm originates from Bayes theorem. Bayes’ theorem helps us calculate the probability of a point of data using history data given that the point falls under a certain class[5]. It can be expressed through the following equation:

P(class|data) = (P(data|class) \*P(class)) / P(data)

Here we are trying to find the probability of the class given the data. Naïve Bayes uses conditional independent assumptions. This means that it assumes that to some point the data points do not depend on each other for occurrence given the class [4]. Naïve Bayes treats all of its attributes equally. “However in real-world applications, different attributes play different roles in discrimination of the class” [3]. The issue is that some attributes are more significant than others, hence a better way for improving the algorithm is by allocating different weights for different attributed. Naïve Bayes theorem can be used for classifying different problems in machine learning. Here are some examples of such problems, determining whether an email is spam or not spam, facial recognition, distinguishing between negative and positive emotions.

Petsche and Staab [6] applied the naïve Gaussian Bayes classification on detector pulses which gives a binary classification problem since the results can either fall in the correct or false detector pulse class. Its likelihood is said to be normally distributed

Other researchers modified the primary naïve Bayes equation to build an even better classifier [10,11]. Tan and Shenoy [10] merged the naïve Bayes method with logistic regression for classification.

# **Methodology**

The course review data provided is in a form of text. The computer is unable to process it in that form. Therefore the data must be converted to binary values. For optimal results before processing text data, it must be ensured that the data has been preprocessed. “Data preprocessing is a data mining technique that involves  transforming raw data into an understandable format [7]. The data is cleaned by removing insignificant words such stop words, characters and numbers. The text must be broken down into smaller pieces of strings. After all the preprocessing has been done the classifier may be built.

In order to build the model the prior and likelihood values must be calculated. From the equation:

Split the data into two sets, the training dataset and the testing dataset. Let S denote the training dataset which is 70 percent of the original dataset,

Where i=1,2,3,…,n is the number of classes and) is the data of different attributes, is the prior information of its probability of appearance in class ,and is the information obtained from the text data itself. is the probability of obtaining the class given the data B.

First calculate the amount of times the attributes fall under a certain class, . Then divide that number by the class total, /A, to find the probability of finding a specific class. Do this for every class. After that, calculate the probability of obtaining a certain class, given the different data attributes.

Find the probability prior, likelihood and the posterior.

# **Analysis**

The data that was provided can be considered as big data. It is in tabular form and has 107018 row and 3 columns. The first column is the numbering of the table. The second column named ‘Review’ contains all the course reviews that were provided by students. The data in this column is of a string format. The third column displays the different classes in which the reviews fall under. These classes range from 1 to 5 whereby 5 means the review is a good review and 1 means the review is a bad review.

The data was cleaned by removing all the stop words and punctuation in the dataset. The importance of this is to avoid creating a model that uses unrelated and insignificant words to classify the reviews. After having cleaned the data, the data got split into training(70%) and testing(30%) dataset. Given that there are a lot of words that are displayed in the dataset, the Multilabel Binarizer was used to convert the reviews into binary digits which are then stored in an array. The label column which displays the different classes was merged into the array to create a new array. To get the prior probabilities of the different classes I first calculated the sum of each class under the column label then divided these sums by the total number of all classes, separately.

To obtain the likelihood, all instances where the label binarizer recorded 1 for the different words in the array were summed up for each class. The product of the sums of the instances provided zero as an output therefore the Laplace smoothing was incorporated in the equation. What it does is that it add 1 to the product to avoid getting zero probability[9, 11]. The product of all the instances recorded as 1 is divides by the total number a class was identified in the label column. The probability of the posterior is the prior probability multiplied by the likelihood. The probability of the a review falling under any of the classes showed to be very low.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

# **conclusion**

Due to the lack of package usage the accuracy of the classifier built is low. Regardless of this, Naïve Bayes plays a huge role in creating models that use text data for classification.

##### **References**

1. I.H Witten, Text mining, (2003)

<https://www.cos.ufrj.br/~jano/LinkedDocuments/_papers/aula13/04-IHW-Textmining.pdf>

1. A. Hotho, A. Nurnberger, G.PaaB, A brief survey of text mining, (2005), <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.447.4161&rep=rep1&type=pdf>
2. S. Chen, G.I. Webb, L. Liu, X. Ma, A novel selective naïve Bayes algorithm, Knowledge-based systems,192 (2020)
3. L. Jiang, L. Zhang, L. Yu, D. Wang, Class-specific attribute weighted naïve Bayes, Pattern recognition, 88 (2019) pp. 321-330
4. J. Brownlee, Naïve Bayes classifier from scratch in python, (2019), <https://machinelearningmastery.com/naive-bayes-classifier-scratch-python/>
5. D. Petschke, T. Staab, A supervised machine learning approach using naïve Gaussian Bayes classification for shape-sensitive detector pulse discrimination in positron annihilation lifetime spectroscopy(PALS), Nuclear inst. And methods in pysics research, 947 (2019)
6. M. Sharma, What steps should one take while doing data preprocessing? (2019), <https://hackernoon.com/what-steps-should-one-take-while-doing-data-preprocessing-502c993e1caa>
7. P. Meesad, P.Boonrawd, V. Nuipian, A chi-square-test for world importance differentiation in text classification, proceedings of computer science and information twchnology, 6 (2011)
8. S. Yang, An introduction to naïve Bayes classifier, towards data science,<https://towardsdatascience.com/introduction-to-na%C3%AFve-bayes-classifier-fa59e3e24aaf>, (2019)
9. T. Tan, P.P. Shenoy, A bias-variance based heuristic for constructing a hybrid logistic regression-naïve Bayes model for classification, international journal of approximate reasoning, 117 (2020) pp.15-28
10. O.F. Arar, K. Ayan, A feature dependent naïve Bayes approach and its application to the software defect prediction problem, applied soft computing, 59 (2017) pp.197-209

**.**