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

Surname: SIBANDA

Name: KUDZAI CARLOS  
Student Number: 201574939

Module Name: ADVANCED MACHINE LEARNING

*Abstract*

*Over the last decade we have seen an increasing interest in the field of text / document classification world and it is more than vital to those involved in machine learning. Correct classification of documents/text is a step closer to data being usable or useful in creating solutions or models that can be used to solve problems. There are a wide variety of classification algorithms and in this paper the use of the Naïve Bayes classifier will be explored.*

Keywords: Sentiment/Document/Text Classification, Naïve Bayes, Probability, Reviews Dataset

# **Introduction**

Sentiment classification is a special type of text classification that became an interesting topic of study in the early 2000s because of a rapid increase in subjective texts from social media, blogs and other sources [24]. There are various terms that are used as alternatives for sentiment classification such as subjectivity analysis, opinion extraction and review mining [27].

In machine learning the methods used in classification are strictly for data processing [15] and allow computers to run multi-class text-classification. They allow the user to place new observations in their categories/classes based on the training sets [8]. There is so much data being created every day because we now live in a digital age and this data comes in different formats; thus, it is important for us to be able to classify the data before attempting to use it. Text classification has been used in spam filtering in emails, ontology mapping etc. [9].

Machine Learning has some algorithms that are used to create classification models such as Support Vector Machines (SVM), Deep Learning and Naïve Bayes. Each of these algorithms work differently from each other [7]. Naïve Bayes classifier is used more due to its simple nature in training as well as classification stage [11]. Naïve Bayes models allow for their attribute to contribute to the final outcome and each independently from the others and that makes it more efficient when it comes to computing [10].

The Naïve Bayes classifier was developed from Bayes’ rule which was formulated by Thomas Bayes (1701 – 1761) and it is structured as follows:



This project will be focusing on using the Naïve Bayes algorithm to classify course review data that was scrapped from Coursera. Naïve Bayes classifiers work on the assumption that the value allocated to each feature independently influences the class and this is referred to as conditional independence and that is where the term ‘Naïve’ was derived [4]. Maitra, Madan, Kandwal and Mahajan expressed that credulous bayes predicts enrollment probabilities for each class, for example, the likelihood that given record or information point has a place with a specific class. The class with the most noteworthy likelihood is considered as the most probable class [16]. It is calculated as follows:

(2)

where Cj is a class and d is an instance

* p(Cj|d) : probability of d occuring in class Cj
* p(d|Cj) : probability of getting d in class Cj
* p(Cj) : probability of getting class Cj
* p(d) :probability of getting d

Despite its simple nature and the assumptions it comes with, it has been proven to be useful in many different domains. It provides practical learning algorithms as well as prior knowledge and pre-observed data can be combined and this brings about an advantage over other classifiers [17] . This project will explore its structure .

1. **PROBLEM STATEMENT**

There has been a rapid growth in the online presence of people and they share their feelings, opinions and feelings [24] and this has come with reviews of products and services. This growth came with a change in the way business is conducted. Reviews can spread very quickly and so it is important for companies to investigate the review data online and use them to improve their service and change their strategies where they are lacking. Sentiment /Document classification is one of the ways in which they can achieve that.

This project will focus in using document classification to deal with review data of an online course to see how the sentiments/ reviews can be categorized.

Given the reviews given by students who took an online course can we be able to analyze the data and categorize it into useful ratings that allow us to further understand how many people are enjoying the course vs the number of people that are not? Using Naïve Bayes classifier can we build a model that accurately categorizes reviews into 5 classes based on their content? What accuracy levels can this produce? Can these results be used to further improve the quality of the content produced in the class?

# **literature Review**

Naïve Bayes classifiers have been used since the late 1990s [12] and they have been developed further and improved since then to suit different types of data. Text mining has become a norm in this age of big data and Naïve Bayes is one of the most used classifiers [9].

Xia, Zong, Li [25] aligned the sources of data that we have which include social media, blogs, forums etc. and how they contribute to sentiment classification.

Quan, Ren [26] outlined the possible benefits of sentiment analysis in areas such as education, opinion polls and e-commerce.

Catal and Nangir [24] performed sentiment classification tests using Naïve Bayes, SVM and Bagging and explored and reported each of their strengths and weaknesses.

Kang [13] conducted experiments in 2012 using Naïve Bayes to conduct sentiment analysis of restaurant reviews. For the performance comparison to be done they also ran SVM and compared the results.

Maitra, Madan, Kandwal, Mahajan [16] used the classifier on faculty feedback data to distinguish between valid and invalid feedback. They were able to build a Feedback Validation Model using Naïve Bayes classifier and this is just one of the educational uses of the classifier.

George Forman [2] explained how feature selection affects the effectiveness of a Naïve Bayes classifier. He also got into feature selection metrics like Bi-Normal Separation and how it improved the accuracy of the model and outperformed the other metrics.

Mita K and Zaveri [3] made a technical review of automatic text classifiers which use a machine learning technique to assign text to pre-defined classes. They further discussed the generic strategy to develop the system as well as the challenges one faces in the process of trying to build the classifier model.

Dong, Shang and Zhu [4] performed tests in an attempt to improve the accuracy of the Naïve Bayes algorithm by adding the improved Gini index algorithm to feature weight and it yielded positive results.

Ting, Ip, Tsang [9] conducted tests to see how well the Naive Bayes classifier performs against other classifiers such as decision tree, support vector machines and neural networks.

Yang and Webb [18] explored the different methods of discretization for Naïve Bayes. Discretization is an approach used to handle numerical attributes in an algorithm. This is done during preprocessing and it improves the performance of the classifier.

# **Methodology**

The data used in this project is a collection of reviews for an online course. The reviews were scrapped from Coursera. There are 107 017 reviews and each of them is labeled from 1 to 5. The rating is from a bad rating(1) to a good rating(5).

The goal of this project is to train a Naïve Bayes classifier using principles of probability theory (likelihood, prior, posterior) in order to perform a review sentiment classification into five classes (rating of 1 to 5)

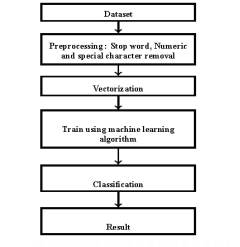


Fig. 1: Proposed approach for classification

Naïve Bayes classifier, like any other classification algorithm, begins with preprocessing. This is the process with which the data is cleaned and prepared for use. When data is collected it is often in an unusable state and contains a lot of attributes that are nor beneficial to the classifier; thus, we remove them.

The first step of this preprocessing phase is using the stopwords algorithm to remove words such as “a” and” the” which do not contribute anything to the performance of the model. Along with removing stopwords we also need to remove periods, punctuations and commas as they also do not add any value to the text when it is being processed [9].

The next step of preprocessing is stemming. This algorithm takes into consideration words that have similar meanings but have differing grammatical form such as “car” and “cars”. The stemming algorithm takes those words and turn them into one attribute and this helps with the attribute correlation of terms in the review dataset we will be working with.

Collect the words which appear more than once in the reviews and store them in a collection called vocabulary. The vocabulary is treated as word sets by matching it to the training data and then collect the probability values of matched sets per class and then the probability can be calculated [19]. The maximization calculation can be utilized to ascertain the likelihood class and afterward all audits can be ordered dependent on which class has the most noteworthy likelihood to it.

Before building our classifier, it is important to binarize the data which means the reviews are converted into numerical vectors so that each review can be numerical and this is achieved by using the MultiLabelBinarizer.

For our classifier to be built we need to consider three things:

1. Posterior probability – this is where one questions the possibility that a particular text can be put into a certain class. In the case of this project it is the probability of a certain review belonging to any of the five classes.
2. Class- conditional probabilities/ likelihood – how likely it is to observe a certain review given a certain class.
3. Prior probabilities / priors – interpreted from prior knowledge using maximum-likelihood estimates.

Maximum-likelihood estimate is therefore calculated as follows:

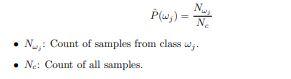


Fig. 2: maximum likelihood estimate

Maximum-likelihood estimates are frequencies of how many time each class appears in the training data [20].

Classifiers use smoothing methods to make adjustmens to the likelihood estimator to make it more accurate [22]. It also helps with the generation of common and non informative words. This project uses Laplace smoothing which was designed to smooth categorical data and deal with the issue of zero probability. When applied to the prior probabilty and probability are denoted as follows:

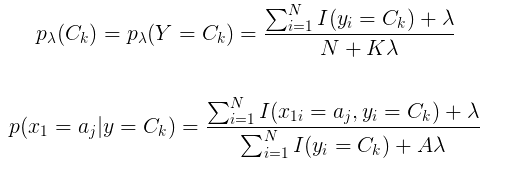


Fig. 3: Laplace smoothing

K is the number of different values in class y and A is the sum of the values in aj. Lambda is usually equal to 1.

“Bengio and Grandvalet said that Cross-validation is an intensive technique which uses all available data as training and testing data. It repeatedly trains the algorithm K times with fraction of 1/K reserved for testing” [23]. In this classification 5 cross validations will be done.

# **Analysis**

The Naïve Bayes classifier was built from scratch. The likelihood was manually calculated manually with an added Laplace smoothing and the priors were manually calculated as well.

The model was built and the first compilation gave an accuracy rate of 74.09 %.

The results from the 5 cross validations were 74 % with very little variation between them (see appendix).

While it is noted that 74% is not as good as one would like to achieve it is however a good result for the first attempt and with a few changes it can be improved.

The model ran successfully and the goal of the model which was to place text into the 5 classes was reached. We can thus conclude that it is possible to carry out that task and also that it can be improved by changing other things to improve the accuracy. The accuracy was not too high in this instance but with more research and learning classifier it could be more accurate.

These classes can thus be used by the administrators of the course to see what they need to improve on as well as what they are doing right. These sentiments can be used to improve the future rating of the service they are providing.

# **conclusion**

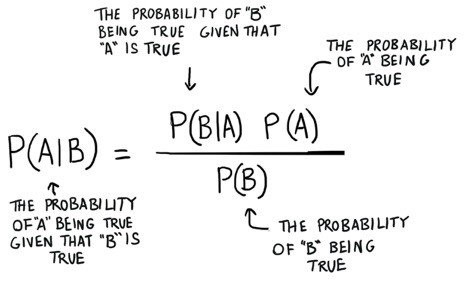
Using sentiment analysis on reviews is a rigorous process however the results allow the user to know where to improve and where to change. In this project we experimented and the results proved that Naïve Bayes classifier is a decent classifier even though a lot can be changed to improve the accuracy. There is a lot more research that needs to go into classifiers, not limited to just Naïve Bayes but other machine learning algorithms such as SVM and Deep Learning as well.

Using cross validation proved useful in improving the results of the classifiers. The possible areas of study are using different preprocessing methods, different feature selections and experimenting with other classification methods.

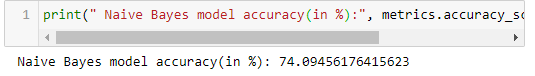
##### **References**

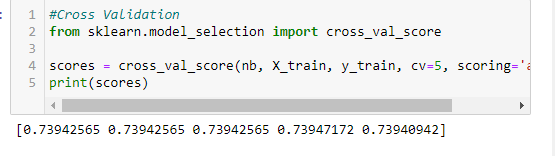
1. Agarwal, B. and Mittal, N., 2014. Text classification using machine learning methods-a survey. In *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012* (pp. 701-709). Springer, New Delhi
2. Forman, G., 2003. An extensive empirical study of feature selection metrics for text classification. *Journal of machine learning research*, *3*(Mar), pp.1289-1305.
3. Dalal, M.K. and Zaveri, M.A., 2011. Automatic text classification: a technical review. *International Journal of Computer Applications*, *28*(2), pp.37-40.
4. Dong, T., Shang, W. and Zhu, H., 2011. An Improved Algorithm of Bayesian Text Categorization. *JSW*, *6*(9), pp.1837-1843.
5. Dai, W., Xue, G.R., Yang, Q. and Yu, Y., 2007, July. Transferring naive bayes classifiers for text classification. In *AAAI* (Vol. 7, pp. 540-545).
6. Tong, S. and Koller, D., 2001. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, *2*(Nov), pp.45-66.
7. Text Classification 2019, viewed 19 July 2020, <<https://monkeylearn.com/text-classification/>>
8. Zhang, Y.C. and Sakhanenko, L., 2019. The naive Bayes classifier for functional data. *Statistics & Probability Letters*, *152*(C), pp.137-146.
9. Ting, S.L., Ip, W.H. and Tsang, A.H., 2011. Is Naive Bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications*, *5*(3), pp.37-46.
10. McCallum, A. and Nigam, K., 1998, July. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization* (Vol. 752, No. 1, pp. 41-48).
11. Chakrabarti, S., Roy, S. and Soundalgekar, M.V., 2003. Fast and accurate text classification via multiple linear discriminant projections. *The VLDB journal*, *12*(2), pp.170-185.
12. Mitchell, T.M., 1997. Machine Learning, volume 1 of 1.
13. Kang, H., Yoo, S.J. and Han, D., 2012. Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews. *Expert Systems with Applications*, *39*(5), pp.6000-6010.
14. Pranckevičius, T. and Marcinkevičius, V., 2017. Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing*, *5*(2), p.221.
15. Alpaydin, E., 2020. *Introduction to machine learning*. MIT press.
16. Maitra, S., Madan, S., Kandwal, R. and Mahajan, P., 2018. Mining authentic student feedback for faculty using Naïve Bayes classifier. *Procedia computer science*, *132*, pp.1171-1183.
17. Narayanan, V., Arora, I. and Bhatia, A., 2013, October. Fast and accurate sentiment classification using an enhanced Naive Bayes model. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 194-201). Springer, Berlin, Heidelberg.
18. Yang, Y. and Webb, G.I., 2002, August. A comparative study of discretization methods for naive-bayes classifiers. In *Proceedings of PKAW* (Vol. 2002).
19. Nigam, B., Ahirwal, P., Salve, S. and Vamney, S., 2011. Document classification using expectation maximization with semi supervised learning. *arXiv preprint arXiv:1112.2028*.
20. Raschka, S., 2014. Naive bayes and text classification i-introduction and theory. *arXiv preprint arXiv:1410.5329*.
21. Tripathy, A., Agrawal, A. and Rath, S.K., 2015. Classiﬁcation of Sentimental Reviews Using Machine Learning Techniques. *Procedia Computer Science*, *57*, pp.821-829.
22. Kaur, G. and Oberai, E.N., 2014. A review article on Naive Bayes classifier with various smoothing techniques. *International Journal of Computer Science and Mobile Computing*, *3*(10), pp.864-868.
23. Bengio, Y. and Grandvalet, Y., 2004. No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research*, *5*(Sep), pp.1089-1105.
24. Catal, C. and Nangir, M., 2017. A sentiment classification model based on multiple classifiers. *Applied Soft Computing*, *50*, pp.135-141.
25. Xia, R., Zong, C. and Li, S., 2011. Ensemble of feature sets and classification algorithms for sentiment classification. *Information sciences*, *181*(6), pp.1138-1152.
26. Quan, C. and Ren, F., 2014. Unsupervised product feature extraction for feature-oriented opinion determination. *Information Sciences*, *272*, pp.16-28.
27. Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, *5*(1), pp.1-167.

**APPENDICES**

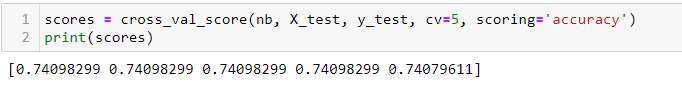
****

**Fig.4 Naïve Bayes**

**Fig. 5 Accuracy readings**

****

**Fig. 6 Cross validation scores (training)**

****

**Fig. 7 Cross validation scores (testing data)**