Comparing Naive Bayes, Logistic Regression, and Random Forest Classifiers on Wine Ratings Analysis

1st Kailen Shinmoto
Natural Language Processing
Occidental College
Los Angeles, United States
kshinmoto@oxy.edu

2nd Rowan Fitch
Natural Language Processing
Occidental College
Los Angeles, United States
rfitch@oxy.edu

***Abstract*—We were given a model that uses Random Forest classifier to analyze the ratings of wines in a database based on the descriptions of each wine, and to do the same but with either a Naive Bayes or Logistic Regression classifier. We chose to look at both options and see which of the classifiers would work best and the differences in their performance. In order to do this we created a new database based on the one given to us that included the wine descriptions along with a cleaned version of these descriptions, their point score, and a simplified version of the point scores putting closely scored wines into groups. The analyses done by our classifiers are based on this database.**

I. Introduction

We were given an example of a Wine Ratings Analysis done by Olivier Goutay [1] that sorted wines into different categories of ratings based on descriptions given of each wine using a Random Forest Classifier. We followed the same process that Goutay did in creating five different qualities to separate the wines into: Underaverage Wines (80-84 pts), Average Wines (84-88), Good Wines (88-92), Very Good Wines (92-96), and Excellent Wines (96-100). Our objective was to copy what the other user did to predict the Wine Rating Analyses but with a different classifier whether it be Naive Bayes or Logistic Regression (the other example used Random Forest Classifier). We chose to look at both types of classifiers and compare the accuracy between all models, including the normal Logistic Regression model and a Multinomial(MN) Logistic Regression. All together we tested four different classifiers: Naive Bayes, Logistic Regression, MN Logistic Regression, and Random Forest Classifier. In order to achieve this we first needed to look at the provided data of the wines.

II. Methodology

A. Preparing The Data

The first step in building our model was to determine which of the features provided to us by the database of wines we would use to analyze the wines into their correct ratings range. We followed the steps that Goutay took and decided that if we are to compare the efficiencies of both Naive Bayes and Logistic Regression to the Random Forest Classifier that Goutay used, it would be best to use the wine descriptions as the feature for our classifiers to analyze.

After deciding this, we then had to clean up our data before being able to analyze the wine descriptions. We followed the steps that Goutay took in his example where he got rid of all wines in the database that were either duplicates or were null values. After doing so, we created a separate database that included only the wine descriptions and the points correlated to those wines. After creating this data base we would create a new column in it to store the simplified point scores (PS) of the wines (1: Underaverage Wine - 5: Excellent Wine). At this point, a sample of our database can be seen in Table 1.

description	Points	PS
”This has great depth of flavor with its fresh ...”	87	2
”Soft, supple plum envelopes an oaky structure ...”	87	2
”This Verdejo smells like citrus fruits and wil...”	90	3
”This is taut and dense, and requires time and ...”	92	4

TABLE I
SAMPLE OF $DataFrame$

After creating this new database, we knew we wanted to clean up our description data more, and decided to remove punctuations from the data. This is one of the differences between our approach to the data and the original approach, where the description data was not altered. We also tried removing stopwords and lemmatizing the data but both of those actions ended up lowering our accuracies so we stuck with only removing the punctuations. In order to do this we first removed the punctuations by replacing all the punctuations with a space and putting the new descriptions into its own separate column.

B. Training the Model

We trained our data on a random data set that was 90% of the total data, this is because Goutay also implemented his wine analyzer using 90% of the total data for training and we wanted to keep our model as close to the example given to us as possible. To implement our Naive Bayes and Logistic Regression models, we used the $sci{-}kitlearn$ Python package. We also used the $CountVectorizer$ object from $sci - kit\,learn$ so that the actual text from the descriptions could be represented in a numerical way. For our Naive Bayes Classifier, we used a Multinomial Naive Bayes Classifier found in the Naive Bayes package of $sci-kitlearn$. This is because of the way the wine analysis is set up being that there are more than two possible ratings the wines can be sorted into. We also used the logistic regression classifiers of the linear model package found in $sci - kit\,learn$, one being a regular logistic regression classifier with its max iterations raised to avoid convergence errors, and a multinomial logistic regression model.

III. Results and Analysis

Naive Bayes	0.74
Logistic Regression	0.90
MN Logistic Regression	0.90
Random Forest	0.96

TABLE II
Model Accuracies

After analyzing the wine descriptions using each of the different classifiers, we found that each of them varied in their accuracies. The Multinomial Naive Bayes Classifier had an accuracy of 74%, both versions of the Logistic Regression models had an accuracy of 90%, and the Random Forest classifier had an accuracy of 96%. The successes of our models can most likely be attributed to the different qualities of the words used in each of the separate wine ratings along with the large portion of the data being used on training our models. However, what are the differences creating the inequalities of accuracy between our models? We believe that the Naive Bayes model had the lowest accuracy because of the way that Bayes' Theorem works. With so many of the wines possibly being close to each other in ratings, it could be easy for some of the wines to have a slightly higher probability score for an adjacent classification compared to its correct one. Logistic Regression is most likely more accurate than Naive Bayes because it takes its probabilities out of a total of 1 unlike Naive Bayes. This means the wines are more likely to be separated into their correct ratings without having to worry about the wrong probability score being higher than the correct one. And lastly, the Random Forest classifier used in the example still worked the best, most likely because of its complex and long process of creating the most accurate decision tree made up of numerous decision trees based on the training data. However, the Random Forest classifier also took the longest to process.

IV. Threats to Validity

Since our classifiers are used from the $sci - kit\ learn$ package, our model is subject to errors and mistakes from the developers of $sci - kit\ learn$. However, given that $sci-kit\,learn$ is a popular package that has been continually updated, this is unlikely.

In this project we were also offered the option to include other features included in the database of wines given to us. After looking at some of the other features, we came to the conclusion that the other features varied too much across the different wine ratings which is why we chose only to implement the wine descriptions as our feature. For an example of this, Figure 1 will show a large spread of prices ranging in the many different sections of wine ratings.

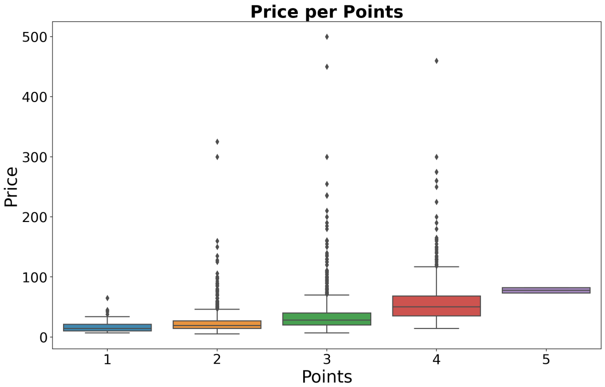

Fig. 1. Range of Prices

V. Conclusion

After analyzing the wine descriptions using each of the different classifiers, we were able to determine the different capabilities in accuracy between the classifiers. We also were able to clean up the data of the descriptions in order to increase the accuracy of all the different models. By cleaning the data and giving each of the classifiers the same training data set, we were able to see the different efficiencies of the classifiers. Our approached focused on these differences in order to determine which of the classifiers worked best, and to possibly theorize why these classifiers worked at different accuracy levels. In the end we determined that the Multinomial Naive Bayes had the lowest accuracy of 74% because of its weakness of being forced to choose the rating with the highest probability, Logistic Regression performed next best at 90% accuracy because its probabilities are derived from a set total of 1, and Random Forest performed best at 96% accuracy because of its complex algorithm made up of numerous decision trees which thus will normally lead to the correct rating analysis of the descriptions.

Related Work

The first article related to our work is "Selection of Important Features and Predicting Wine Quality Using machine Learning Techniques" [2]. This article focused on exploring the use of machine learning techniques like linear regression, neural network and support vector machine for product quality. First, they determined the dependency of target variable on independent variables and secondly, they predicted the value of the target variable. They used linear regression is used to determine the dependency of target variable on independent variables, and used neural network and support vector machine to predict the values of the dependent variable. Although not exactly related to what we did, it was interesting to look at another machine learning project that looks at the features of wines.

The second article related to our work is ”A Message Classifier Based on Multinomial Naive Bayes for Online Social Contexts” [3]. This article focuses on messages sent online by minors and classifying messages, as normal or dangerous in the case of sexual harassment, according to the risk they present to the minor. They do this by integrating a Multinomial Naive Bayes classifier to analyze the messages sent online and determine whether they are normal or dangerous. They chose to use a Multinomial Naive Bayes model because of the possibility that certain words can be included in both safe and dangerous contexts. We found this related to our work because of its use of a Multinomial Naive Bayes classifier where there are numerous outcomes to think about.

References

[1] Olivier Goutay, ”Wine Ratings Analysis w/ Supervised ML”, https://www.kaggle.com/olivierg13/wine-ratings-analysis-w-supervised-ml, 2018

[2] Yogesh Gupta, ”Selection of Important Features and Predicting Wine Quality Using machine Learning Techniques”, December 2017

[3] Thársis Salathiel de Souza Viana, et al, “A Message Classifier Based on Multinomial Naive Bayes for Online Social Contexts”, April 2018