Laptop-Scale Text Analysis: Can Product Description Be Used to Predict Customer Ratings?

Elsa Li
elli@g.hmc.edu

Abstract

The way a person perceives a scent could differ wildly from that of another person. However, there are many perfumes that are almost universally loved or hated. As the use of social networks helps perfume makers reach more people with their official descriptions, it is important to ask if there is an underlying psychological effect that these descriptions have on the customer buying the product. By training Multinomial Naïve Bayes, Complement Naïve Bayes, and Logistic Regression Classificiation models to predict customer review ratings, from 0.00 to 5.00, from product description, I found that the Multinomial Naïve Bayes model performs most accurately out of the three models. However, this model only slightly outperforms two dummy classifiers. Therefore, I conclude, at least from the aforementioned three models, that perfume description is not a good predictor of perfume rating.

1 Introduction

I started off this project by trying to answer the question: Why are some terrible perfumes receiving high ratings? I wanted to understand if there were other factors, in addition to the actual smell of the perfume, that impacted the way customers viewed their purchase. I wanted to see if certain aspects of the perfume metadata impacted the perfume ratings. My final question is: How does product description impact customer ratings? To answer this question, I used a perfume data set from the website Fragrantica, which featured a diverse range of perfumes, their scent profiles, and real customer reviews and ratings (joehusseinmama, 2024). I isolated the product description and rating features and ran a vectorizer to convert the text to numbers. Then, I used the Multinomial Naïve Bayes, Complement Naïve Bayes, and Logistic Regression Classificiation models from Scikit-Learn library to predict the perfume's rating from its description. I also compared the results of the different models to see which one does best in its prediction of the rating, given the description.

2 Related Work

One way that previous research had looked at rating prediction was by examining the sentiment behind each review. Though this was slightly different from my question, it was still applicable since it is looking at consumer behavior. One paper proposes a new algorithm called Sentiment Fuzzy Classification to improve sentiment analysis for movie ratings. Essentially, the algorithm blurs the line between different sentiments and uses different shapes to classify sets of words into fuzzy sets. Strangely, the authors of the Fuzzy Classification algorithm did not specify their accuracy and precision in their paper. Rather, they discussed the general idea of them and why they're important to data science (Mouthami et al., 2013).

A similar study identifies the Logistic Regression Classification (LRC) model as the best classification model, compared to Multinomial Naïve Bayes and Linear Support Vector models. To evaluate the models, the author tested their accuracy on an Amazon reviews dataset. Similar to the last paper, this paper used customer reviews to predict the product's rating. By comparing with the true ratings, the authors found that all the models performed better than random chance selections, and that LRC did better than all other models with an accuracy of 54.1%. However, the authors did not state how their models performed compared to a model that would classify everything into the category that appeared most (Taparia and Bagla, 2020).

Another related study examines how customer reviews changes over time, and proposes a "dual-channel framework" neural network for predicting ratings using series data about the dynamic reviews. The neural network intakes the stream of reviews,

and updates its review prediction based on every new review that it gets fed. The author also ran their model against popular models such as a Bi-directional LSTM with Attention model, and their model outperforms most of them (Zhang et al., 2023).

Although all three studies examine how to predict ratings from reviews, there was not a lot of papers that looked at how a product's seller-written description could be impacting people's reviews. I believe this may be because a lot of papers are looking at more tangible items that have a more polar definition of good and bad, unlike perfumes.

Therefore, this let me to wonder what would happen if I replaced the reviews in these studies with the product descriptions, and how that would affect the accuracy of the respective models.

3 Dataset

I used a dataset titled "Fragrantica Data" from the website Kaggle. The original dataset contains 6 different files, 4 of which being CSV files, with the other two being an image and a README document. Out of the 4 CSV files, one was the complete version, and the other three were subsets of that CSV file. Therefore, for completeness, I decided to use the complete CSV file for this data analysis project.

According to the author of the dataset, the dataset is scraped information from Fragrantica, a popular perfume review website, and contains data about around 84,000 unique perfumes. The columns of the dataset represent the name, designer, product description, 80 most recent customer reviews, link to the perfume on Fragrantica, and a number rating, from 1.00 through 5.00, of the perfume.

With Prof. Xanda's guidance, I also made sure to verify the source of the dataset. There had been a previous dataset of women's clothing reviews which initially seemed to be credible. However after searching online, I could not find the sources of the reviews. Thankfully, after a quick search on a few of the perfume descriptions, I was able to see that they were posted on Fragrantica by real humans.

Since my research question focused on ratings, it was helpful to understand how the rating spanned in the whole dataset. Therefore, with the provided feature in Kaggle, I visualized the rating ranges of the dataset.

Rating Ranges	Number of Perfumes
0.00 - 3.00	64
3.00 - 3.40	137
3.40 - 3.80	445
3.80 - 4.20	1067
4.20 - 4.60	557
4.60 - 5.00	193

Clearly, the vast majority of ratings are above 3, with most of the perfumes receiving a rating between 3.40 and 4.20, with some perfumes that received a higher rating of 4.20 and above. Therefore, I will implement a similar sort of split of the data once I start cleaning the data.

4 Methods

4.1 Data Cleaning

Something of note is that some of the perfume names contained non-English languages. Since I did not want to complicate the models with having to train on a variety of languages, I decided to exclude any non-English content from the cleaned data. I did so by running the langdetect library on each description, and only adding the perfume's description to the an array of descriptions if it was detected to be English.

When creating the dataset to train and test on, I exclusive extracted the description and the corresponding ratings columns from the original dataset. Then, I created three buckets to store the ratings. For ratings that ranged from 0.00 to 3.40, they were placed in Bucket 0, which was the low bucket. For ratings that ranged from 3.40 to 4.20, they were placed into Bucket 1, which was the medium bucket. For ratings that ranged from 4.20 to 5.0, they were placed into Bucket 2, which was the high bucket. I created these splits after examining the general distribution of the ratings. Then, to a corresponding array of ratings, I added the aforementioned numerical representation of the bucket that rating of the perfume fell into.

Then, I trained the model based on description as the input and bucket number as the predicted output.

4.2 Running Models

I decided to start by using a Multinomial Naïve Bayes (MNB) model, since that was the model that everyone in the group started off with. Before I could use the model to predict the rating, I had

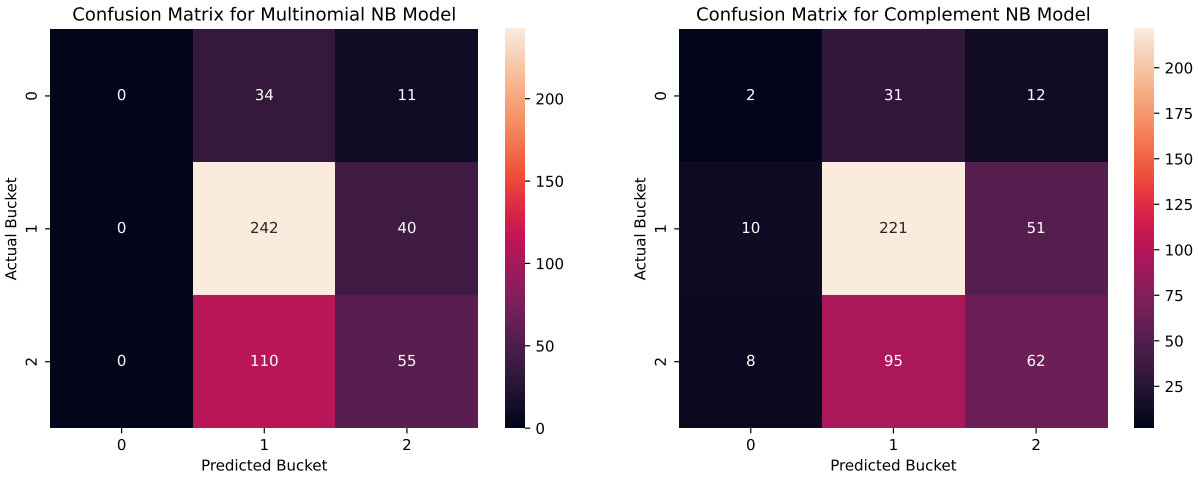

Figure 1: Left: Confusion matrix for the MNB model, Right: Confusion matrix for the CNB model

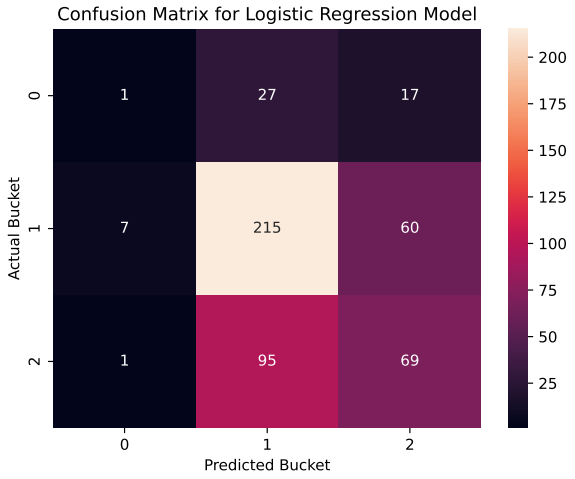

Figure 2: Confusion matrix for the LR model

to first convert the description text into vectors of textual features. I used Scikit Learn's Count Vectorizer function to convert the description text into a numerical matrix. Then, I fitted the vectorized descriptions with the MNB model. I fed the same training and test set to the MNB model.

Next, I tried out a Complement Naïve Bayes (CNB) model since it was a similar algorithm, and I was wondering if it might make any significant improvement in the precision and accuracy. I fed the same training and test set to the MNB model.

Using the results of the related work, I also decided to try out the Logistic Regression (LR) model. Again, I decided to lean on Scikit Learn's robustness, and utilized the library's provided LR model, again using the same training and test data as the MNB and CNB models.

5 Results

5.1 Accuracy Values

From the outputs of the MNB model, I found that the accuracy was 60.37%. For comparison, a dummy classifier that randomly places perfumes into buckets without considering their description would have an accuracy of 33.33%, since there are three different buckets. After removing the non-English occurrences, I had 2458 unique perfumes, with the majority, 1430 perfumes, being sorted into Bucket 1. Therefore, a dummy classifier that sorts all perfumes into the most common bucket would have an accuracy of 58.18%. The MNB model did much better than the random sorting model, and only very sightly better than the model that sorts everything into the most frequently appearing bucket.

After feeding in the exact same training and test

sets, the accuracy of the CNB model turned out to be 57.93%, which is not a significant difference from the accuracy of the MNB model. With the accuracy of the CNB model, it outperforms the random sorting model, but loses to the frequency sorting dummy classifier.

The LR model had a similar story. Surprising, It performed exactly the same as the CNB model, with an accuracy of 57.93%. While it does better than the randomization classifier, it still does worse than the frequency classifier.

5.2 Confusion Matrices

The confusion matrices of the models were unimpressive as well. While all three models did well in predicting perfumes that were in Bucket 1, none were very precise with its prediction.

Something of note is that MNB model, which outperforms both the CNB and LR models, chose to not predict any of the input as being from Bucket 0. While the other two models did not predict a significant amount of perfumes to be from Bucket 0, this could still be a contributing factor in the MNB model's slightly superior performance. Another possible factor for MNB model's better performance could be because it predicted the most perfumes to be from the most frequent bucket, Bucket 1. In total, MNB predicted 386 perfumes to be from Bucket 1, while CNB predicted 347 perfumes and LR predicted 337 perfumes. The MNB model might have been able to take advantage that most of the perfumes fall into Bucket 1, and used this fact in conjunction with its avoidance of predicting Bucket 0 to perform better than both types of dummy classifiers.

6 Discussion

Disappointingly, the accuracy of the models show that it is difficult to reverse engineer a good rating by writing a convincing product description. Since the best model out of the three models only slightly outperforms the dummy classifier, it is hard to conclusively say that any of these models were helpful in predicting the perfume's rating, given the perfume's description. This was a little surprising since I expected there to be more of an impact, since I personally believed that a good product description made me more likely to think of the perfume in a good light, even before I purchased it. This led me to think about the potential causes for the disconnect between description and rating. I believe a potential cause could be that a significant amount of time passes between when a customer reads the description of the product, and when they leave their rating. Thus, no matter how the manufactures had written their product descriptions to be, the customer simply do not remember once they sit down to rate the product.

Another potential reason that I was unable to create an accurate prediction model could be because of how people's opinion shift before and after they receive their purchase. One possible situation is as follows: The customer reads a very convincing product description, and is compelled to purchase the perfume. However, once they receive their purchase, they decide that the perfume does not live up to their expectations. Therefore, a better research question for next time could examine whether product description help raise the chance that a potential customer makes the purchase.

References

joehusseinmama. 2024. Fragrantica data.

K Mouthami, K Nirmala Devi, and V Murali Bhaskaran. 2013. Sentiment analysis and classification based on textual reviews. In *2013 international conference on Information communication and embedded systems (ICICES)*, pages 271–276. IEEE.

Ankit Taparia and Tanmay Bagla. 2020. Sentiment analysis: predicting product reviews' ratings using online customer reviews. *Available at SSRN 3655308*.

Xin Zhang, Linhai Zhang, and Deyu Zhou. 2023. Sentiment analysis on streaming user reviews via dual-channel dynamic graph neural network. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7208–7220.