Classifying Sentiments on the Amazon Fine Food Reviews Dataset

Tales Ivalque Taveira de Freitas, NLP, INSPER

***Abstract*—The abstract goes here.**

***Index Terms*—IEEE, IEEEtran, journal, LaTeX, paper, template.**

I. Dataset

THIS document presents the classification of sentiments on the Amazon Fine Food Reviews dataset. The dataset contains thousands of reviews of various food products, each consisting of a note of 1 to 5, a plain review text, and some other features. This dataset was used in the paper (1), which looks into how the previous experiences affects the grades given by each user, and how that can be used in a classification model.This dataset span a period of more than 10 years, including all 500,000 reviews up to October 2012.

II. Classification Pipeline

The classification model chosen for this project was the Bernoulli Naive Bayes(2) algorithm, which is particularly well-suited for binary features, such as word presence in text. Unlike other versions (like Multinomial Naive Bayes, which counts occurrences of features), Bernoulli Naive Bayes uses binary features (1 for presence, 0 for absence). As a Naive Bayes variation, this classifier is based on Bayes' Theorem and assumes that all features are independent, which is often referred to as a ”naive” assumption.

For this dataset, the chosen feature was the Text, which consists of plain-text reviews of food products, and the target was Score. For the target, we transformed numerical scores into categorical labels: scores of 3 or below were labeled as ”Negative,” and scores above 3 as ”Positive.”

The Text feature underwent several transformations to ensure optimal performance for the classifier:

1) **Cleaning**: Removed all non-alphanumeric symbols, leaving only words and numbers.
2) **Stopword Removal**: Common words (e.g., ”this”, ”that”) which do not carry significant weight were removed to ensure meaningful words contribute to the model.
3) **Stemming**: Words were reduced to their root form (e.g., ”running” became ”run”) to treat variations of the same word uniformly.
4) **Text Augmentation**: Techniques such as synonym replacement were used to artificially expand the dataset, improving model robustness and generalization.

Following preprocessing, we built a classification pipeline that consisted of:

- **CountVectorizer**: Converts the text into a bag-of-words representation.
- **TfidfTransformer**: Transforms word counts into Term Frequency-Inverse Document Frequency (TF-IDF) values.
- **Bernoulli Naive Bayes Classifier**: Classifies the text as positive or negative based on the binary word presence features.

Figure 1 shows a diagram of the pipeline used in this project.

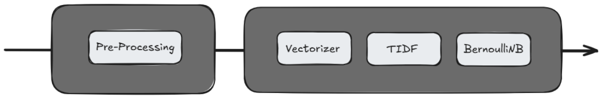

Fig. 1. Diagram of the pipeline used in this project.

III. Evaluation

In this step, we evaluate the classifier by splitting the dataset into training and test sets multiple times, ensuring that each split is shuffled to avoid biases. Since the dataset is not perfectly balanced, we use the balanced accuracy score 3 as the primary evaluation metric, which accounts for imbalanced class distributions.

Figure 2 shows the uneven distribution of the Target Feature

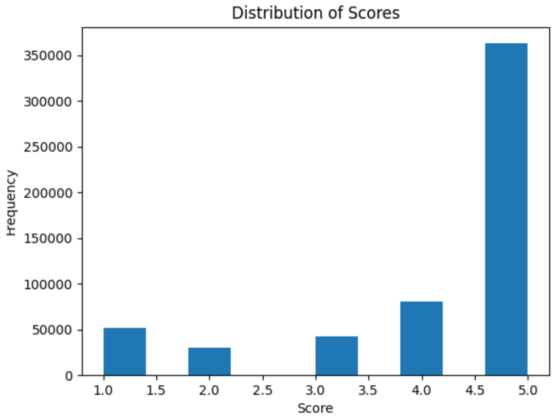

Fig. 2. Histogram with Score distribution on the original dataset.

As the data was not given with prior separation, in the Training step, it was necessary to split the test and train sets from the full dataset. This separation was made by shuffling the dataset and then splitting the dataset 80/20 for train and test, respectively.

Then, using the pipeline shown on the last section, with no change to the default parameters we trained the model, then we evaluated its performance on the test set using the balanced accuracy score to account for class imbalance. The classifier achieved a balanced accuracy of 0.72, indicating reasonable performance across both positive and negative sentiment classes.

The classification report, including precision, recall, and F1-score, is summarized in Table 1. The model performed better in predicting positive sentiments, as indicated by higher precision, recall, and F1-score for the positive class.

TABLE I
CLASSIFICATION REPORT FOR SENTIMENT ANALYSIS

Class	Precision	Recall	F1-score	Support
Negative	0.63	0.53	0.57	24,930
Positive	0.87	0.91	0.89	88,756
Accuracy	0.83 (on 113,686 samples)			
Macro avg	0.75	0.72	0.73	113,686
Weighted avg	0.82	0.83	0.82	113,686

In addition, we analyzed the most important words contributing to the classification decisions. Table 2 lists the top 10 most important words, ranked by their importance scores. These words were identified using the log-probability differences between the two classes (positive and negative).

TABLE II
TOP 10 MOST IMPORTANT WORDS FOR CLASSIFICATION

Word	Importance Score
nonmoney	-6.3757
chiou	-4.9074
mistakesbr	-4.7784
parse	-4.7663
abattoir	-4.7038
tobacman	-4.6710
refundable	-4.6020
productslower	-4.5656
carcinogenicbr	-4.5656
billswhen	-4.5656

IV. DATASET SIZE

We evaluated the effect of dataset size on model performance by downsampling the dataset from 10% to 100% and measuring the classifier's performance at each step. For each fraction, the data was shuffled, split into training and test sets, and evaluated over 10 iterations using the balanced accuracy score.

Analyzing the results, it is shown that it will be very difficult to improve our model solely by increasing the dataset size. This is primarily due to the significant imbalance in the distribution of target values. Therefore, it would be more beneficial to focus on gathering additional negative reviews to improve our model's performance.

Figure 3 shows the result of this analyse.

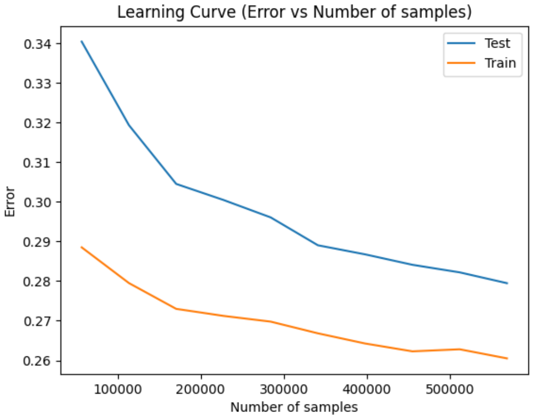

Fig. 3. Mean error for each number of samples.

V. TOPIC ANALYSIS

As seen in TABLE I, The Positive classification shows a significantly better accuracy. As there are only two values for our Target, and no other column on our original dataset that would be useful as a target, there will not be a re-evaluation of accuracy using a two-layer model.

REFERENCES

[1] J. McAuley and J. Leskovec., "From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews," *WWW*, 2013.

[2] D. Ceccon. (2019) Tipos de métodos naive bayes. IA Expert Academy. Accessed: 2024-10-03. [Online]. Available: