ISyE 6740 – Summer 2024
Final Project Report
Business Reviews Sentiment Analysis

Andrea Ruiz, Garrett McClellan, Ranjodh Sandhu

Background

The business services industry has undergone significant transformation in recent years, driven largely by the increased convenience of researching and accessing services online. This digital shift allows clients to explore options and make decisions from the comfort of their homes or any location.

Yelp is a popular online platform where users rate and review local businesses, such as restaurants, based on their experiences. This data-driven approach helps users make informed decisions, but it relies heavily on individual reviews, which can be subjective and varied. Yelp currently employs features to highlight the most helpful reviews, however, as the comments and ratings given are subjective and some people weigh certain factors more than others, there is room for ambiguity. This can result in a mismatch between the rating given to a business and the reviews themselves and create mistrust.

Furthermore, sentiment analysis is a process that involves analyzing text to determine the sentiment expressed within it (classify). In the context of platforms like Yelp, sentiment analysis can be used to assess the overall sentiment of user reviews about a business.

Problem Statement and Objective

Reliance on online reviews is important since clients can't physically assess businesses beforehand. Reviews and ratings are key to judging service quality and reliability, but they're subjective and can lead to mismatches between text and star ratings. This inconsistency undermines trust and complicates decisions for potential clients.

The primary objective is to ensure that subjective sentiments in Yelp reviews match the star ratings users give. This alignment is crucial for enhancing the credibility and accuracy of online reviews. To address this challenge, this project will leverage Natural Language Processing (NLP) and Machine Learning (ML) techniques to analyze and develop a strong system that connects the emotional tone and wording in Yelp reviews with the star ratings given.

Data Source

We loaded the Yelp dataset, which is a comprehensive resource that includes a vast array of user-generated content from Yelp’s platform. It contains a wide range of business categories, from restaurants and bars to salons and repair services. The dataset includes: roughly 6.900,000 user reviews and data information from 150,000 businesses. We focused on the reviews json file in particular which contains the data below.

review_id	Unique ID of the review
user_id	Unique ID of the user
business_id	Unique ID of the business
stars	ratings of the business (1-5)
date	review date
text	Text review from the user
useful	number of users who vote a review as useful
funny	number of users who vote a review as funny
cool	number of users who vote a review as cool

Table 1: Aspects of the .JSON data files

Methodology

To analyze the Yelp review dataset, several key steps were undertaken to gain insights into customer sentiments and behaviors. Initially, the dataset was loaded from a JSON file and converted into a pandas DataFrame for structured analysis. In addition, basic statistical summaries were derived to understand the distribution characteristics of numerical variables such as star ratings and review votes (cool, funny, useful). Figure 1 depicts the raw data, top 20 users, their average rating and number of reviews, the star review distribution throughout the data. This helps to show some of what the raw data conveys.

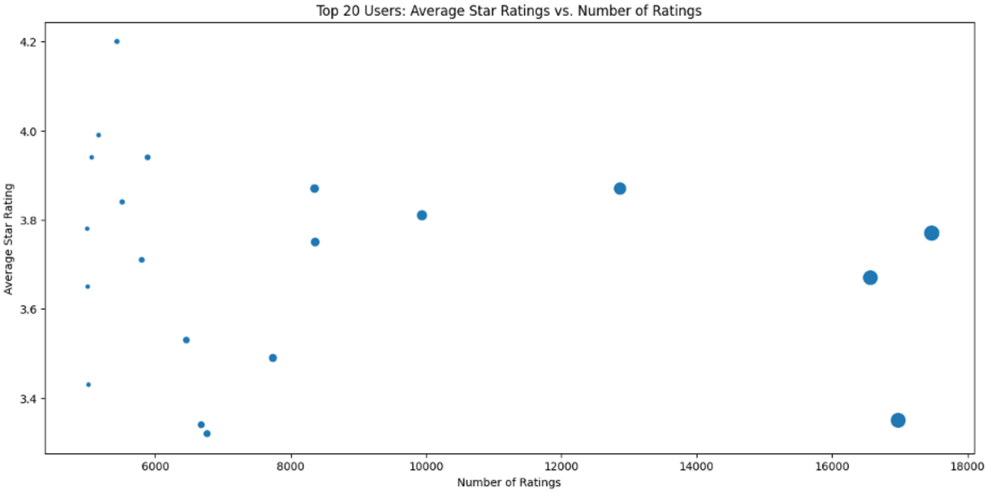

Figure 1: This figure shows different users average rating **and** ***number of reviews***

Preprocessing text data is essential for modeling because it transforms raw text into a format suitable for our prediction models. In a script we performed essential preprocessing tasks, including tokenization, lowercasing, stop words removal, and lemmatization, prepare the review texts for analysis. The preprocessed data is then saved for subsequent steps.

For model training and evaluation, we used a data classification script to train various machine learning models, Linear SVC (Support Vector Classifier), SGD Classifier (Stochastic Gradient Descent Classifier), Logistic Regression, and Naive Bayes, on the preprocessed data. To test and train each of these models we used 20/80 split respectively. We chose these four models because Linear SVC, Logistic Regression, SGD Classifier, and Naive Bayes are ideal for NLP tasks due to their ability to handle high-dimensional data, efficiency in computation, scalability, robustness to sparsity, and proven performance in text classification. We were initially going to test a random forest model as well, but due to computing power limitations, we decided to test just the four models listed above and were able to get all models to run and thus get results regarding each of the four models.

Each classifier offers unique advantages for sentiment analysis. Linear SVC excels in handling complex relationships between features and sentiment labels, making it effective even when data is not easily separable. SGD Classifier is optimal for large datasets due to its fast-training capabilities, which is crucial for processing extensive amounts of text data efficiently. Logistic Regression calculates the probability of sentiment classes using a logistic function, providing a clear interpretation of how features contribute to sentiment prediction. Naïve Bayes leverages TF-IDF values or word counts to probabilistically determine sentiment class probabilities, making it robust for classifying sentiment based on text features. TF-IDF (Term Frequency-Inverse Document Frequency) assigns weights to words based on their frequency and importance across documents, helping classifiers like Naïve Bayes to discern the relevance of words in sentiment analysis. Each model was evaluated using metrics such as accuracy, precision, recall, and F1 score on the test set, offering a comprehensive comparison of their performance in sentiment prediction tasks. The results are as shown in Table 2.

Model	Accuracy	Precision	Recall	F1 Score
Linear SVC	0.6848	0.6519	0.6848	0.6553
SGD Classifier	0.6390	0.5860	0.6390	0.5603
Logistic Regression	0.6972	0.6745	0.6972	0.6809
Naïve Bayes	0.6136	0.5471	0.6136	0.5403

Table 2: Results from analyzing different ML models vs review text and estimating the star rating given

After finding that the Logistic Regression model was the most accurate model for this data, we took steps to further enhance model performance with the developed logistic regression model algorithm. To manage computational demands, we chose a 30/70 split for training and testing data instead of the typical 80/20 split. This model uses

Randomized Search CV (Cross-Validation) to fine-tune the Logistic Regression model's hyperparameters. Specifically, we focused on tuning the hyperparameters {'solver': 'newton-cg', 'C': 1} using Randomized Search CV. This technique systematically explores various combinations of hyperparameters within specified ranges. The 'solver': 'newton-cg' parameter in Logistic Regression refers to the Newton-Conjugate Gradient algorithm, minimizing the cost function using gradients and Hessian matrices. The Hessian matrix helps determine the direction and rate of convergence during model training by iteratively adjusting model parameters based on the gradients of the cost function. The regularization parameter 'C': 1 controls the strength of regularization: a smaller value increases regularization to prevent overfitting but may underfit. These settings balance model complexity and performance in logistic regression, ensuring robust convergence and accuracy.

The results, including review IDs, actual sentiments, and predicted sentiments, are saved for further analysis. By following these steps, our methodology ensures a thorough analysis of review sentiment, leveraging advanced natural language processing and machine learning techniques to align subjective sentiments with objective star ratings, thereby enhancing the reliability and trustworthiness of online reviews.

Results and Evaluation

In this section, we analyze the performance of our logistic regression model in predicting star ratings based on user reviews. We assess various evaluation metrics, including histograms, box plots, ROC curves, precision-recall curves, and confusion matrices, to understand how well the model captures and predicts different star ratings.

Figure 2 shows a histogram comparing the distribution of actual stars given by users and the stars predicted by the model. This visualization reveals a notable trend: the model tends to overestimate both 1-star and 5-star ratings while underestimating the more moderate ratings of 2, 3, and 4 stars. Before finalizing the model, we anticipated a more balanced distribution of predicted stars, expecting that the model would temper extreme ratings to produce more moderate predictions. However, the current results indicate that the model accentuates extreme ratings rather than mitigating them. This suggests a need for further refinement of the model to better capture and predict the more nuanced ratings (two, three, and four stars) and to provide more reasonable and reliable reviews.

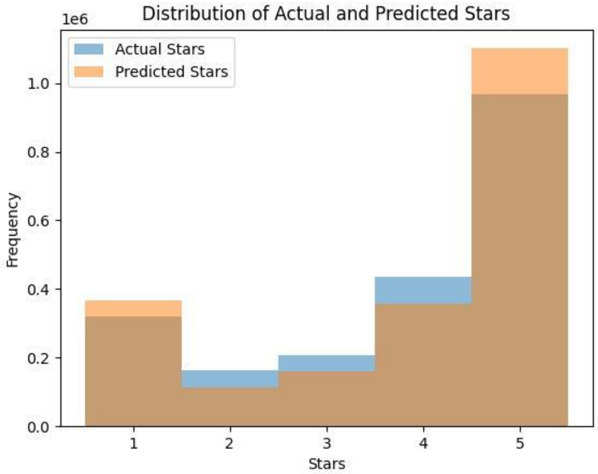

Figure 2: Distribution of given stars and model predicted stars

Figure 3 illustrates a box plot of the predicted star ratings for each actual star rating. The box plot highlights the spread and central tendency of the predicted ratings for each actual rating level. Interestingly, the variance at the extreme ends (1-star and 5-star ratings) is considerably less than that of the middle ratings. This pattern indicates that while the model can consistently predict extreme ratings, it struggles with accurately predicting the more moderate reviews, which exhibit a wider spread and greater variability in predicted ratings.

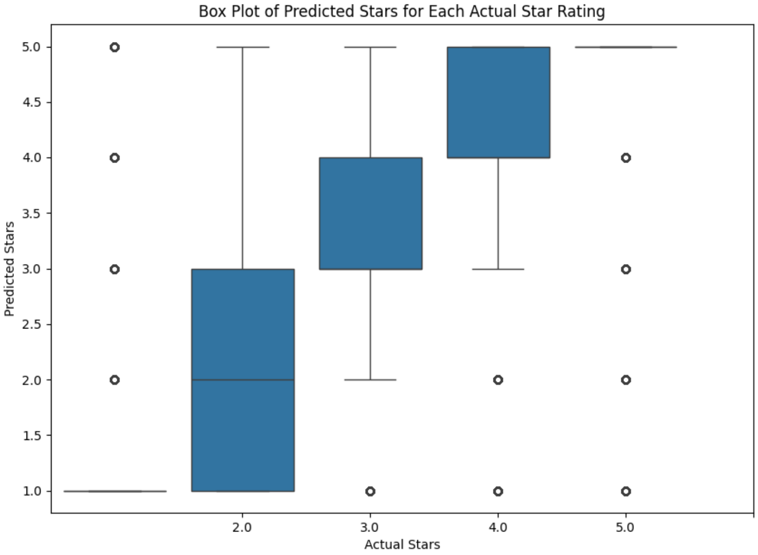

Figure 3: Box plot to show the predicted stars vs actual

Figure 4 presents the ROC curves for the Logistic Regression model, showing the true positive rate versus the false positive rate for each star rating. The ROC curve helps in understanding the performance of the model across different thresholds. The area under the curve (AUC) values for each star rating demonstrate varying levels of prediction accuracy. For instance, the AUC for 1-star reviews is 0.90, indicating high model performance in identifying 1-star reviews. Conversely, the AUC values for 2-star, 3-star, and 4-star reviews are around 0.65 to 0.67, which signifies moderate performance and highlights the areas where the model needs improvement to distinguish between these moderate ratings.

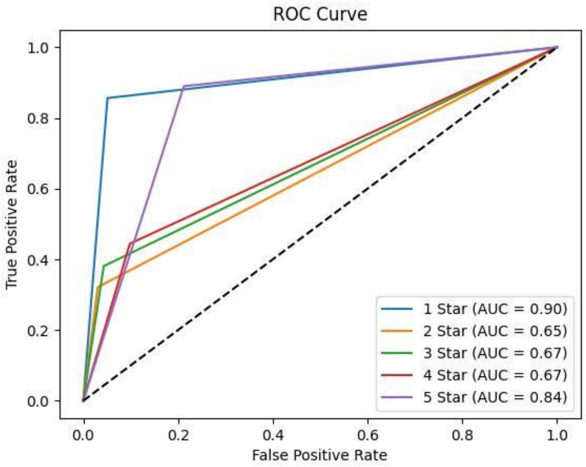

Figure 4: ROC Curve of the Logistic Regression Model

Figure 5 shows the precision-recall curves for the Logistic Regression model, plotting precision against recall for each star rating. Precision-recall curves are particularly useful for understanding the trade-offs between precision and recall. The curves indicate that the model achieves high precision and recall for 1-star and 5-star reviews, suggesting it can effectively identify the most negative and most positive feedback. However, the precision and recall drop significantly for the middle ratings (2-star, 3-star, and 4-star), reaffirming the need for enhanced model training to improve its performance on moderate reviews.

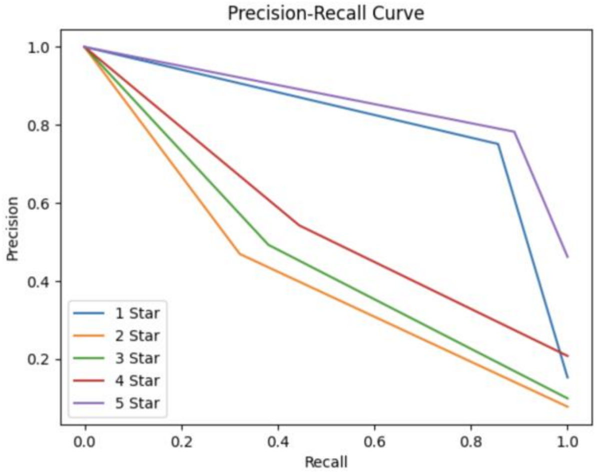

Figure 5: Precision-Recall Curve of the Logistic Regression Model

Checking the top features for each star rating is important for understanding the different aspects of customer feedback. By analyzing the most frequently mentioned words for each rating level, from 1.0 stars to 5.0 stars, we can identify what drives both positive and negative sentiments in the reviews. This process helps us to pinpoint specific issues or highlights that influence the overall rating. The top features of 1.0-star Yelp reviews are all negative, with terms like "worst," "horrible," and "terrible" being the most common. These words show that reviewers are very dissatisfied. For example, "worst" appears most frequently, indicating it is a major complaint. Other words like "poisoning," "rude," and "unprofessional" highlight specific issues such as poor service or safety problems. To improve, businesses should focus on fixing these key issues. Comparing these negative terms with those in higher-rated reviews can help identify what is lacking and guide improvements.

Top 10 features contributing to 1.0 stars:	
worst	**11.4763**
horrible	**8.4319**
terrible	**7.8702**
poisoning	**7.8045**
zero	**7.0481**
disgusting	**6.8382**
rude	**6.5478**
awful	**6.5246**
waste	**6.4872**
unprofessional	**6.3278**

Table 3: Top features contributing to 1 star reviews

In contrast, the top features of 5.0-star reviews are very positive. Words like "amazing," "delicious," and "excellent" are most common, showing that reviewers are extremely happy with their experiences. "Amazing" has the highest frequency, indicating it is a major term in positive reviews. Other terms like "fantastic," "awesome," and "best" also emphasize the high quality of experiences. Understanding these positive features helps businesses know what is appreciated and continue to enhance these aspects to maintain high customer satisfaction.

Top 10 features contributing to 5.0 stars:	
amazing	**11.8114**
delicious	**11.7574**
excellent	**9.2564**
great	**9.0208**
perfect	**8.5463**
fantastic	**8.4609**
awesome	**8.4002**
best	**8.2721**
perfection	**7.9452**
phenomenal	**7.5343**

Table 4: Top features contributing to 5 star reviews

The introduction of the confusion matrix for true versus predicted star ratings was helpful as well to understand the chosen logistic regression model’s performance. This matrix compares the actual star ratings with the ratings predicted by the model. The corner of the matrix where both true and predicted ratings are 5 stars, this corner has the highest number, it signifies that the model has a strong ability to correctly identify 5-star reviews, which is a key indicator of high recall and precision for the top rating. Our confusion matrix shows that the model excels at predicting both 1-star and 5-star reviews, with the darkest colors and highest numbers in these corners. This means it effectively identifies the most negative and most positive feedback. However, it struggles with intermediate ratings, 2-star, 3-star, and 4-star reviews, where the colors are lighter and numbers lower, indicating lower precision and recall. This highlights areas where the model needs improvement. Including this analysis in the report is important because it clearly shows the model's strengths and weaknesses and helps tune our current model.

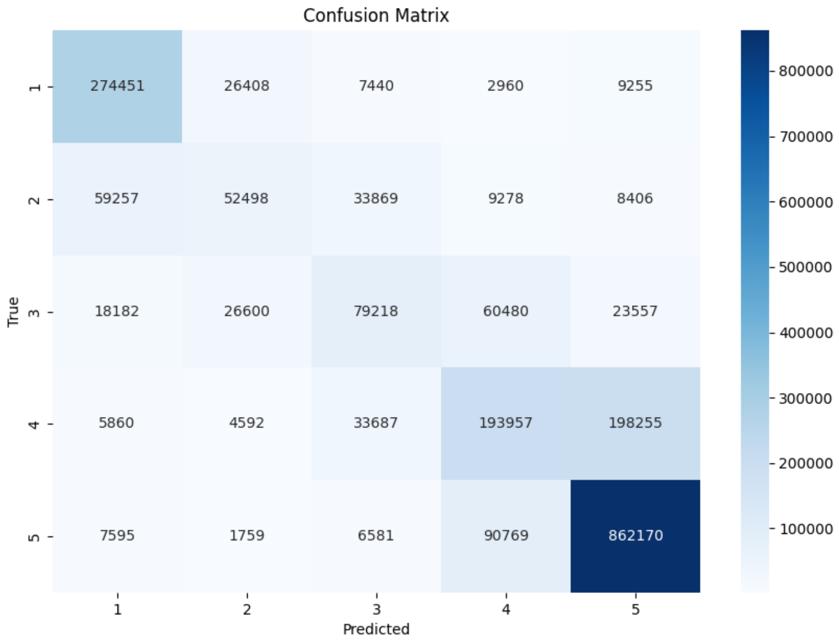

Figure 6: Confusion Matrix for the Logistic Regression Model on the Yelp data

Another aspect of this project that aids in understanding the data and the outcomes of the models is the examination of example predictions of review sentiments for both correct and incorrect guesses. We also see how preprocessed text becomes a predicted star rating and how accurately this prediction matches the actual given review, as shown in the table below.

Review Text	Actual Stars	Predicted Stars
went lunch found burger meh . obvious focus burger amount different random crap pile flavor meat . burger patty seemed steamed appeared preformed patty , contrary stated menu . get ground beef kroger make burger blow water	2.0	2.0
needed new tire wife 's car . special order next day , dropped morning work called hour later car ready . quick efficient , woman helped awesome .	5.0	5.0
nothing beat pizza beer book . place nail , eye towards sustainability always make feel good . specially brewed beer , handful craft beer tap satisfy almost taste . prefer deep dish , although accomplice like thin crust . usually mean leftover . wish appetizer adventurous , changed , guess n't mess success 're pizza place apply . wish delivered .	5.0	4.0
went mom couple week ago . recently went gluten-free , sweet christine 's nearby life saver ! donut blondie muffin (n't judge !) . donut n't worth . small , awkwardly sweet , weirdly textured .. n't need life . would n't get . muffin essentially chocolate chip muffin . best muffin 've ever life . granted , n't muffin	4.0	5.0

two month , love muffin , gluten free option like sign god care u gluten intolerant folk , . wo n't able go back july , , 'm stocking . n't care price . muffin best .		

Table 5: Preprocessed text and its given and predicted star rating

Since the model performs well with extreme ratings but struggles with moderate ones, a practical solution is to integrate direct user feedback. Targeted surveys or interviews with users who give 2, 3, or 4-star reviews can uncover specific issues or preferences that the model might miss. This additional feedback will enhance understanding of customer concerns, leading to more accurate predictions and improved decision-making.

Conclusion

Our study aimed to better match subjective sentiments in Yelp reviews with their star ratings. We found that Logistic Regression was the most accurate model, with a 69.72% accuracy rate. However, all models had only modest accuracy, showing that understanding nuanced sentiments in text is challenging. In addition, the confusion matrix showed that the model performs well for extreme ratings (1-star and 5-star) but struggles with intermediate ratings (2-star, 3-star, and 4-star). This means the model accurately identifies the most positive and negative reviews but has difficulty with the middle ratings. The challenge lies in the intricate nature of sentiment analysis, where interpreting subjective language in reviews isn't straightforward.

To replicate and build on this work, start by obtaining a dataset like Yelp's, including user reviews, star ratings, and metadata. Then, preprocess the text by tokenizing, lowercasing, removing stopwords, and lemmatizing. Train machine learning models like Logistic Regression, Linear SVC, SGD Classifier, and Naive Bayes, and evaluate them using metrics such as accuracy, precision, recall, and F1 score. Analyze the results with a confusion matrix to see how well the models predict both extreme and intermediate ratings. To improve accuracy, refine the model by adjusting hyperparameters. Applying the improved model to new review data will provide useful insights for businesses to enhance their services. This method lays a strong foundation for others aiming to achieve better accuracy and deeper insights in sentiment analysis.

Despite these complexities, our method, which included thorough assessment metrics like precision, recall, and F1 score, establishes a reliable framework for improving sentiment analysis in online reviews. Future work could focus on refining how we preprocess text or exploring advanced models to boost accuracy and applicability in other online reviews.

Work Breakdown

All the three team members contributed equally to this project.

Andrea Ruiz contributed to the report writeup and explanation of code and the preprocessing and exploration of yelp dataset.

Garrett McClellan contributed to the report writeup and the initial exploration of the yelp data in figure creation and preprocessing.

Ranjodh Sandhu contributed by developing the majority of all the model development and results, as well as the preprocessing and exploration of yelp dataset.

References

Arora, S. (2024, June 6). *Sentiment analysis using Python*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/07/sentiment-analysis-using-python/

Wikimedia Foundation. (2024, June 22). *Receiver operating characteristic*. Wikipedia. https://en.wikipedia.org/wiki/Receiver_operating_characteristic

Brownlee, J. (2020, August 27). *Tune hyperparameters for Classification Machine Learning Algorithms*. MachineLearningMastery.com. https://machinelearningmastery.com/hyperparameters-for-classification-machine-learning-algorithms/

Ambarish. (2018, March 9). *A very extensive data analysis of yelp*. Kaggle. https://www.kaggle.com/code/ambarish/a-very-extensive-data-analysis-of-yelp

Brownlee, J. (2023, October 10). *How to use ROC curves and precision-recall curves for classification in Python*. MachineLearningMastery.com. https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/

Yelp dataset. (n.d.-b). https://www.yelp.com/dataset/documentation/main