SIADS 696 Milestone II Project Report

Project Title: Restaurant Customer Review Analysis

Team Members: Sangram Sanjiv Malekar, Minyan Gao, Maggie Oliver

Introduction

Customer reviews have a significant impact on the success of a restaurant and influence potential customers. Our project aims to perform a comprehensive analysis on restaurant customer reviews using both supervised and unsupervised approaches and techniques to gain insights from our datasets. We will focus on three distinct tasks:

1. Sentiment classification analysis of restaurant reviews
2. Extracting popular food/drink items using named entity recognition
3. Topic modeling to uncover topics and themes within the reviews

Our first task of sentiment classification (supervised learning) wants to analyze and classify restaurant reviews into five star ratings. We will explore four different models, which are Naive Bayes, Decision Trees, Random Forests, and Logistic Regression, before choosing the one that achieved the highest accuracy, the Logistic Regression model. The model was further optimized by testing various solvers, penalties, regularization strength (C hyperparameter), and weight. The best logistic regression model had these hyperparameters: solver = “lbfgs,” penalty = “l2,” C = 10, and class_weight = None. A feature importance analysis will be performed to understand the association between top features (words) and their star ratings. We will also be conducting a failure analysis to understand the weaknesses in our model to prevent and mitigate.

Our second task of extracting popular food/drink items (supervised learning) will develop a NER classifier to categorize words in the review text data as “food,” “drink,” or None. We will do this using a baseline Conditional Random Fields (CRF) model and a LSTM model. Various regularization techniques will be utilized to enhance the LSTM model performance and the best values for each hyperparameter are: learning rate of 0.0001, weight decay (L2 penalty) of 1e-05, and a dropout rate of 0.4. A sensitivity analysis will be conducted to understand how changes in the parameters can affect the performance of the model. We will also be performing a failure analysis to identify vulnerabilities in our model.

Our final task of topic modeling (unsupervised learning) will explore and utilize three algorithms to help uncover topics. The algorithms are: Latent Dirichlet Allocation (LDA) employed, Latent Semantic Indexing (LSI), and Non-Negative Matrix Factorization (NMF). Coherence scores will be employed to evaluate the quality and coherence of the topics. The NMF model is indicated to be most effective in uncovering topics based on it achieving the highest overall coherence score of 0.739 for topics.

By understanding customers’ preferences through the analysis of their reviews, restaurants can make informed changes to improve how they operate to attract and satisfy customers.Our project desires to bridge the gap between customer feedback and actionable insights for restaurants and contribute to long term success in a competitive market.

Related Works

We found three projects that share similarities to what we are proposing: “Sentiment Analysis of Restaurant Reviews,” “Restaurant Review Analysis Using NLP and SQLite,” and “Topic modelling on BBC news article.” While both “Sentiment Analysis of Restaurant Reviews" and “Restaurant Review Analysis

Using NLP and SQLite" aim to analyze their datasets regarding restaurant reviews, they differ from each other by using different datasets and their objectives.

“Sentiment Analysis of Restaurant Reviews" seeks a prediction model to categorize restaurant reviews as positive or negative. This is similar to our sentiment analysis, however, we will be categorizing them as star ratings from 1 star to 5 stars. Our project also includes Named Entity Recognition (NER) and topic modeling. “Restaurant Review Analysis Using NLP and SQLite" aims to help restaurant owners understand which food items of their restaurant are liked and disliked by creating a model using restaurant reviews. This aspect is similar to the NER part of our project. Instead of using a given dataset like our project, this project also includes the collection of data where customers can directly input their reviews.

“Topic modeling on BBC news article” analyzes 2,225 BBC’s news articles and seeks out the top five words talked about within each of the five article categories: business, entertainment, politics, sports, and tech. While the data we are working with are completely different, our project also aims to extract latent topics. We will be working with restaurant reviews to answer what most customers like or dislike about the restaurant.

Overall, all three projects share aspects that are similar to parts of our project. However, we seek a more thorough and comprehensive understanding of customer restaurant reviews on various factors, including good/bad reviews through sentiment analysis, finding out liked and disliked food items, and popular reasons why customers like or dislike the restaurant through topic modeling.

Data Sources

There were two data sources used for this project. The primary dataset was the Yelp Dataset and the external data set was 10000 Restaurant Reviews Dataset from kaggle. Details on both data sources are discussed below.

Primary: Yelp Data

Our primary dataset will be using the review dataset from the Yelp Dataset. This dataset is sourced from Yelp Inc., an American company that collects and publishes user reviews and recommendations about businesses ranging from restaurants to entertainment. The Yelp Dataset is a subset of Yelp’s businesses, reviews, and user data for use in academic research. It encompasses 6,990,280 reviews, 150,346 businesses, and over 1.2 million business attributes such as star ratings, hours, parking availability, ambience, etc.

We will be primarily using the review dataset, which is a JSON file that is accessible and downloadable here after extracting. It contains the full review text data along with its user_id to identify who wrote the review and the business_id which is the business the review is written for.

External: 10000 Restaurant Reviews Data

We are planning to also use the 10000 Restaurant Reviews Dataset, which is a dataset of 10000 restaurant reviews from 100 restaurants. It includes restaurant names, reviewer names, their reviews, star rating, time, etc. It is a CSV file that is accessible and downloadable here. This will improve the accuracy of NER and give more reviews and items for NER to extract.

Feature Engineering

Preprocessing Data

Given the large volume of Yelp's review dataset, we decided to sample data to a specific subset to ensure manageability for supervised and unsupervised tasks. By filtering by the total number of reviews for each city in Yelp's business dataset, it was discovered that the city with the most reviews was Philadelphia. We narrowed our review data down to only businesses within Philadelphia city (about 600,000 reviews), which is suitable for our analysis.

Before doing any analysis on restaurant reviews, it is important to preprocess the raw review text data and remove any noise. We've done this through several steps, such as removing punctuation, lowercasing, tokenization, removing stopwords and stemming. After transforming and cleaning the data, it is stored as a pickle file so it can be easily accessed for further analysis. It will be used for supervised and unsupervised tasks, including sentiment analysis, named entity recognition (NER), and topic modeling.

Feature Extraction

For sentiment analysis, a bags-of-words approach was used for feature representation. After tokenizing the review text into individual words, each word was treated as a feature and given a binary value. Review token dictionaries along with their associated star ratings were created where the keys are the word tokens and the values are set as True. These words are the features for the sentiment analysis machine learning model. For our second supervised learning task, we used Named Entity Recognition (NER) techniques, such as pycrfsuite.ItemSequence for Conditional Random Fields (CRF) and vector embedding for LSTM, to help us identify and extract food/item entities from the review text data.

For topic modeling, we employed the 'TfidfVectorizer' to generate a table, document-term matrix, from the raw corpus of restaurant reviews. This table shows how often each word appears in each review compared to how often it is in all the other reviews. Using these features, we are able to do further analyses which we have described in more detail in their respective supervised and unsupervised sections of the report.

Part A: Supervised

In the supervised portion of our project, we have two tasks we are :

1. Sentiment Classification Analysis of Restaurant Reviews
2. Extracting popular food/drink items using Named Entity Recognition (NER)

Details for each task will be discussed in their respective sections of the report below.

Sentiment Classification Analysis of Restaurant Reviews

Description

Our project's goal is to develop a model that accurately predicts the sentiments in the restaurant reviews in Philadelphia city. By categorizing reviews into sentiment categories, we aim to understand the customers' experiences with restaurants for businesses to improve their customer satisfaction.

Data Source

The Yelp dataset, the primary dataset, was used for the sentiment analysis. Details for the data source was explained in the Data Source section of the report.

Methods and Evaluation

We experimented with four different types of algorithmic models using Sklearn implementation and chose the one with the highest accuracy to further optimize. The four types of models and our reasonings for choosing them are below:

1. Naive Bayes - a probabilistic model based on Bayes' theorem with assumption of feature independence
    - They are commonly used for text classification and would be useful in our objective of sentiment analysis on restaurant reviews.
2. Decision Trees - a tree-based algorithm that is excellent at classification and regression tasks
    - They are suited for text classification as they handle numerical and categorical data well.
3. Random Forests - an ensemble method that combines multiple decision trees to make predictions
    - They are good at handling large datasets and effective at capturing relationships between features and their labels.
4. Logistic Regression - a statistical model
    - They are often used for binary classification tasks like sentiment analysis.

We evaluated each of the models' performance by using accuracy as a metric. The algorithms were compared based on their accuracy to select the one that performed best for further optimization. We used a 5-fold cross validation for the mean metrics and their standard deviations.

Model	Accuracy	Standard Deviation
Naive Bayes	0.50526	0.02493
Decision Trees	0.42971	0.00766
Random Forests	0.54223	0.01635
Logistic Regression	0.59996	0.01398

Figure 1: Comparison Table of Naive Bayes, Decision Trees, Random Forests, and Logistic Regression for Sentiment Analysis

As seen in Figure 1, the Logistic Regression model achieved the highest accuracy among the classifiers we experimented with. The parameters of the Logistic Regression model were further optimized by exploring different solvers and their corresponding accuracies. The solvers we tested are Limited-memory Broyden-Fletcher-Goldfarb-Shanno (lbfgs), A Library for Large Linear Classification (liblinear), Newton Conjugate Gradient (newton-cg), Stochastic Average Gradient (sag), and Stochastic Average Gradient Accelerated (saga).

Through comparison, *lbfgs* achieved the highest accuracy of 0.60032 among the solvers explored. We utilized *lbfgs* and continued to fine-tune the Logistic Regression model with different penalty types, regularization strengths, and class weights. The best performing configuration for the Logistic Regression model is with the *solver = 'lbfgs,' penalty = 'l2', C = 10, class_weight = None* with an accuracy score of 0.60115. The precision, recall and f-scores were also calculated along with their standard deviations and presented in Figure 2 below.

Performance Metrics	Scores	
	Macro	Standard Deviation
Accuracy	0.60115	0.01410
Precision	0.58814	0.01449
Recall	0.60115	0.01410
F-score	0.59065	0.01526

Figure 2: Performance Metrics Table for Logistic Regression Model

Using the optimized logistic regression model, we were able to categorize reviews into star ratings ranging from 1 star to 5 stars.

Feature Importance

For this model, features and labels were extracted from the training set and then converted into feature vectors using the TF-IDF vectorization technique. A pipeline was built using the TF-IDF transformer, chi-squared feature selection, and our best logistic regression classifier from earlier. The top, relevant features to our task were selected using the SelectKBest method with the chi-squared function. The logistic regression model learned and understood the relationship between the selected features and their star ratings from the training data.

After fitting the model to the training data, we extracted the coefficients of the selected features. The coefficients are log-odds ratios of the star ratings variable given the frequency of each word feature in the reviews. A higher magnitude of the coefficient indicates a stronger association between the word (feature) and its star rating (target variable). This can be seen in our visualization, Figure 3, that showcases the top twenty features (words) that affect star ratings strongly.

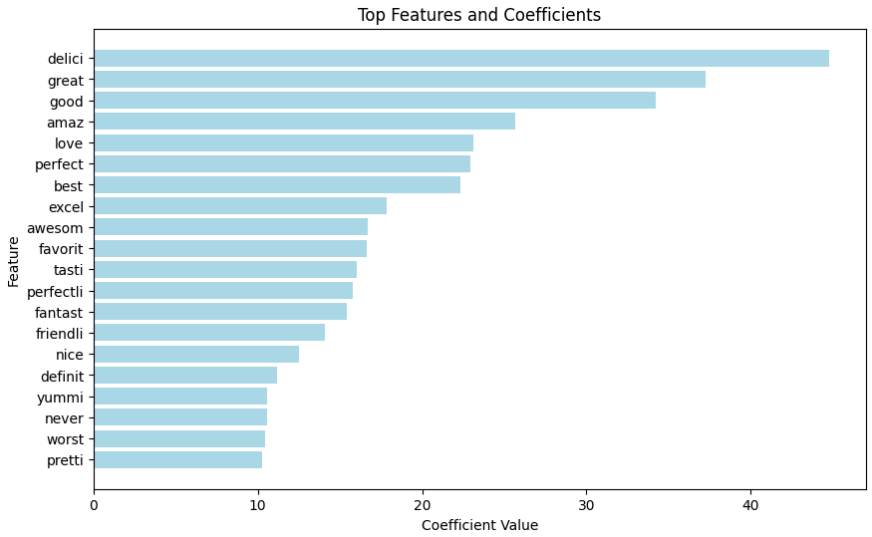

Figure 3: Feature Importance Visualization for Top 20 Features and Coefficients

Failure Analysis

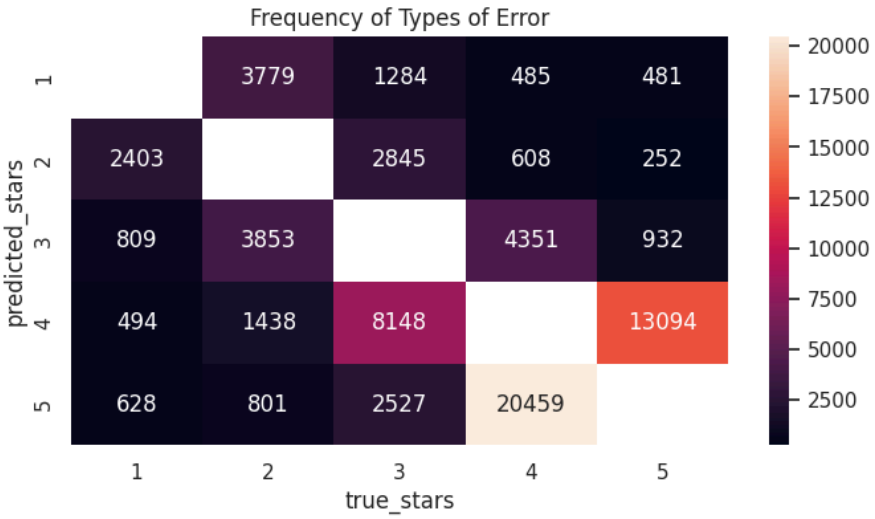

Figure 4: Frequency of Types of Errors Heatmap Visualization

Figure 4 shows the frequency of errors of different types for the final logistic regression model. The chart demonstrates that the model does not seem to trend towards predicting a higher star rating than a review's actual rating more often than lower, or vice-versa. Instead, there are a similar amount of errors where the predicted star rating is too high and too low. The most common types of errors we see are those where the predicted rating is one star above or below the actual rating, especially where the predicted rating is five and the actual rating is four, and where the predicted rating is four and the actual rating is five. Errors where the predicted star rating is very far from the actual star rating, like reviews that are predicted to have five stars but actually have one, are much more rare. One possible reason that so many reviews are predicted to have one more or less star than they actually do is that similar star ratings may have a lot of features in common with each other. The similarity of features between groups likely contributes to the errors we are seeing. One way to improve the model to decrease the occurrence of these errors may be to do more rigorous feature engineering and use fewer features in the model.

Extracting popular food/drink items using Named Entity Recognition (NER)

Description

We aimed to create a Named Entity Recognition (NER) classifier with the goal of categorizing terms in review text into three distinct classes: 'food', 'drink', or 'None'. Such a NER model was used for a downstream task to extract popular food and drink items from thousands of reviews based on the frequency of occurrence of such items. For compute power, GPU-enabled environments provided by the Great Lakes Cluster were used to execute computationally intensive workloads such as hyperparameter tuning, Language Model inferences, etc.

Data Source

The Yelp dataset, the primary dataset, was used for this task. Details for the data source was explained in the Data Source section of the report.

The Yelp and Kaggle datasets lacked annotations for review text, rendering it unsuitable as ground truth data for training purposes. To address this limitation, we employed a synthetic approach to create ground

truth annotations. Leveraging Language Models, specifically Microsoft's language model Phi2, we classified terms within the review text into NER labels. This process enriched the dataset with accurate NER labels, thereby facilitating the development of NER classifier models.

The model was trained and tested using the Yelp dataset, while evaluation was conducted using the Kaggle dataset. The training and testing datasets from Yelp provided the necessary samples for model development and validation, while the Kaggle dataset served as an independent evaluation set to assess the model's generalization performance.

Methods

We implemented the Conditional Random Fields (CRF) model as a baseline model. CRF is a statistical model that considers surrounding text terms when predicting the label for a target term. Our model took into account the term preceding and succeeding the current term to make NER label predictions.

Features for each term in the review is structured as follows:

- The current term and its corresponding Part-of-Speech (POS) tag.
- The previous term and its POS tag.
- The next term and its POS tag.

Further, RNN architecture was adopted to develop a LSTM based model. Each term within the input review text underwent a conversion process into an integer representation utilizing a maintained word-to-integer lookup table. These integer representations were vectorized using PyTorch embeddings. The vectorized terms were sequentially fed into the LSTM model, where at each step, both a hidden state and a cell state were generated. The cell state retained contextual information from previous terms, enabling the model to capture longer-term dependencies within the text. The hidden states were utilized to predict the NER label for the current term. A fully connected (FC) layer with three output variables corresponding to the three NER labels ('food', 'drink', 'None') was applied to the hidden states. These outputs were transformed into label probabilities via the softmax function, with the label possessing the highest probability being assigned as the final predicted label.

To enhance the performance of the LSTM NER model on unseen data, various regularization techniques were implemented (dropout rate, L2 regularization on the loss function, and learning rate optimization). The model underwent fine-tuning with these hyperparameters and the following values were obtained:

- Learning rate: 0.0001
- Weight decay (L2 penalty): 1e-05
- Dropout rate: 0.4

Additionally, the model's performance on the test dataset reached a plateau after approximately 50 epochs, indicating the convergence of training.

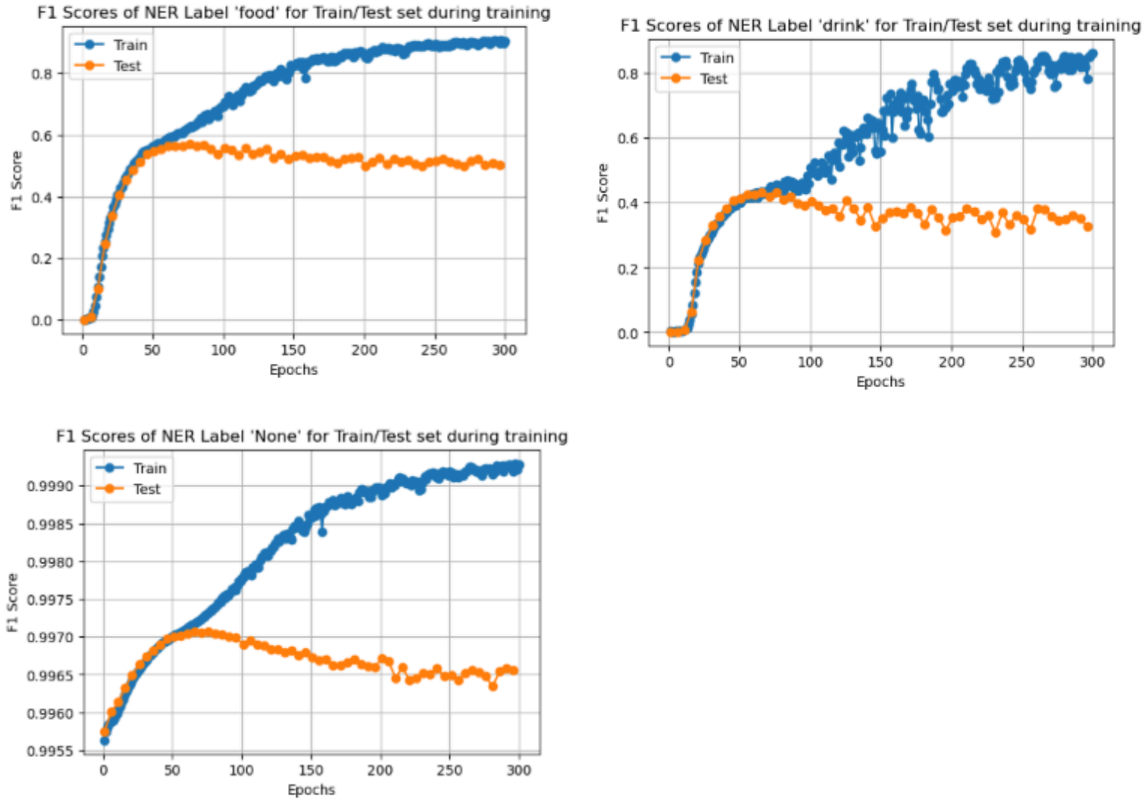

Figure 5: Three graphs of F1 Scores for each NER Labels: food, drink and None

Evaluation

For evaluating the performance of our models, we selected precision, recall, and F1 score (harmonic mean of precision and recall) metrics, aiming to minimize misclassifications of labels. Given the dataset's imbalance, with more instances of 'None' labels compared to 'food' or 'drink', the classifier might perform well on 'None' labels but exhibit lower accuracy on other labels. Consequently, we opted to evaluate the individual F1 scores for each label and compute the macro average (treating all classes equally despite the unequal representation of 'food/drink' labels in the dataset) instead of micro average, to derive the overall F1 score. Below in Figure 6 and 7 are the performance metrics calculated from the test dataset for each respective model.

Performance Metrics	Scores per NER Label from LSTM model		
	None	Food	Drink
Precision	1.00	0.69	0.57
Recall	1.00	0.43	0.33
F1-score	1.00	0.53	0.42

Figure 6: Performance Metrics Table for LSTM Model

Performance Metrics	Scores per NER Label from CRF model		
	None	Food	Drink
Precision	0.96	0.65	0.57
Recall	0.98	0.47	0.31
F1-score	0.97	0.54	0.40

Figure 7: Performance Metrics Table for CRF Model

Macro-averaged F1 scores obtained for Conditional Random Fields and LSTM are 0.64 and 0.65, respectively.

Sensitivity Analysis

The macro-averaged F1 score exhibits sensitivity to weight decay (L2 penalty), learning rate and dropout rate. Higher weight decay and dropout rates leads to model underfitting due to excessive regularization, resulting in poorer predictions. On the other hand, increasing the learning rate has demonstrated a positive impact on performance improvements.

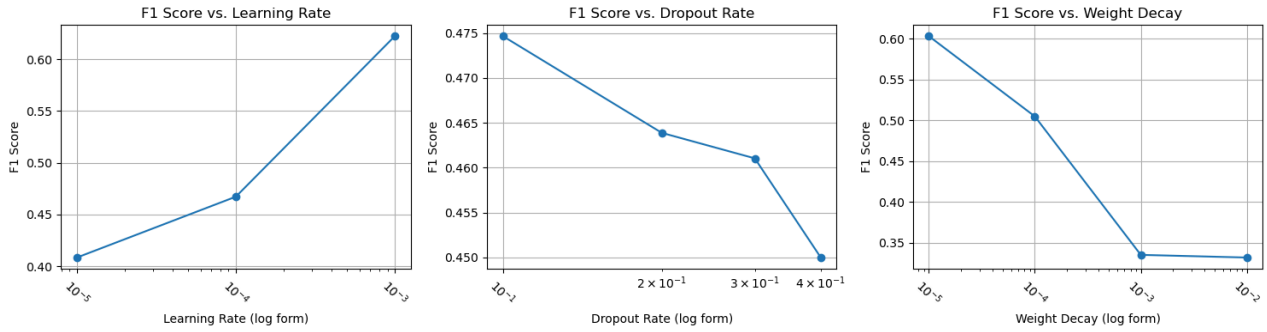

Figure 8: Three graphs comparing F1 score to Learning Rate, Dropout Rate, and Weight Decay

Failure Analysis

We ran LSTM model to predict labels on previously unseen reviews from Kaggle and fetched 3 reviews where NER label predictions were incorrect:

1. Snippet from review #32: "*SICILIAN PIZZA - The pizza was stuffed a lot with toppings, cheese was a bit less*". Model prediction: {'food': ['pizza', 'pizza'], 'drink': []}. Model should have labeled 'SICILIAN' and 'cheese' as food as well.

2. Snippet from review #6777: "*We orderd banjara kebab which was extremely soft, juicy and outstandingly tasty.*". Model prediction: {'food': [], 'drink': []}. Model should have labeled 'banjara kebab' as a food item.

3. Snippet from review #4001: "*Starters are very good to start with mango flavour prawns adds classic mango taste.*". Model prediction: {'food': ['mango', 'mango', 'pizza', 'pizza'], 'drink': []}. Model should have labeled 'prawns' as a food item as well.

Figure 9 shows the confusion matrix for predicted vs. true labels on the test dataset:

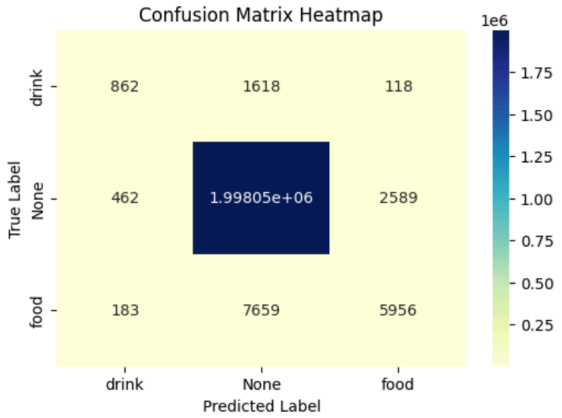

Figure 9: Predicted vs. True Labels Confusion Matrix Heatmap

The mislabeling observed in the reviews above can be attributed to insufficient training data and the presence of potential false positives and false negatives in the ground truth labels. Mislabeled ground truths, as labeled by the Microsoft Phi2 language model, present a significant concern. Predictions generated by Phi2 may contain instances of false positives or false negatives in both the training and test sets, where 'food/drink' items are incorrectly tagged as 'None' and vice versa. Consequently, the NER model's performance may suffer due to such errors in ground truth labels in datasets. Additionally, there is a need to upsample the minority classes (food, drink) to improve F1 score.

Trade Offs

During the evaluation of both of our supervised learning tasks, we encountered an important trade off between the size of the training data and model performance accuracy. While using the entirety of our large dataset would lead to better performances for both the sentiment analysis model and the NER model, it would also be incredibly time-consuming. As mentioned earlier in the Preprocessing Data section of the report, we opted to sample data only from businesses in Philadelphia city. This ties into another tradeoff which is between model training time and accuracy. By sacrificing the size of the training data, we expedited the model development effort. However, as mentioned earlier, that may lead to lower model performance and accuracy compared to using the full dataset.

Part B: Unsupervised

Description

Topic modeling can provide interesting latent themes from document corpus. Such analyses on customer reviews can aid in making informed strategic decisions for businesses and enable them to tailor their offerings and services to better meet customer needs and expectations.

Data Source

We conducted topic modeling on Yelp restaurant reviews to uncover latent themes. Our analysis focused on a prominent restaurant in Philadelphia, selected based on its abundance of reviews, to delve into the prevailing themes surrounding its offerings, ambiance, service, and more.

Methods

We sought to extract meaningful topics from the review corpus by leveraging following algorithms:

1. LDA (Latent Dirichlet Allocation): which assumes that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution of words. It is suitable for initial exploratory analysis.
2. LSI (Latent Semantic Indexing): which achieves topic modeling through Singular Value Decomposition (SVD) by reducing the dimensionality of the document-term matrix.
3. NMF (Non-Negative Matrix Factorization): which factorizes the document-term matrix into non-negative matrices, resulting in a sparse and non-negative representation of the original corpus.

We transformed the raw review text into a document-term matrix using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer. This matrix encapsulated the frequency of terms across the entire review corpus. Subsequently, we applied the LDA, LSI, and NMF algorithms to this matrix to discern the top N topics prevalent within the restaurant's review data.

These algorithms decomposed the document-term matrix into 2 components: document-topic matrix and the topic-term matrix. The topic-term matrix is then utilized to extract top words associated with each topic. On the other hand, the doc-topic matrix enabled the retrieval of documents with the highest weights for a given topic, thereby aiding in the understanding of topic distributions across corpus. This approach enabled us to gain insights into the most discussed themes pertaining to food, cuisine, ambiance, location, service quality, and other relevant aspects.

Evaluation

In addition to human interpretation, we employed coherence scores to assess the quality and coherence of the identified topics. Coherence scores quantify the semantic similarity among terms within a topic. We fine-tuned the hyperparameter representing the number of topics. Hyperparameter tuning revealed that NMF employing seven topics yielded the highest overall coherence score of 0.739. Below in Figure 10 are the word clouds representing the 7 topics generated by NMF.

Figure 10: Word Clouds generated for Topic 1 through Topic 7

Above topics exhibit coherence, but there are few topics that bear notable similarities. These can be interpreted as follows:

Topic 1 and 2 suggest that the restaurant is located in a bustling area near a terminal.

Topic 3 indicates that the restaurant is renowned for its pork roast sandwiches.

Topic 4 suggests that the restaurant is famous for its fresh selection of meat and seafood produce.

Topics 5 and 6 imply that it is a fantastic place with numerous positive reviews.

Topic 7 suggests that the restaurant offers excellent desserts and ice cream.

Also, the review # 4099 has the highest weight for topic 3:

"*Place is awesome. Something for everyone. Dinic's Roast pork sandwich is my favorite!*"

Sensitivity Analysis

Sensitivity analysis was conducted by varying the number of topics across different values to identify the optimal number of topics that yields the maximum overall coherence score. The coherence score reached its peak at 3 topics for LSI, at 14 topics for LDA, and at 7 topics for NMF, as illustrated in below line charts.

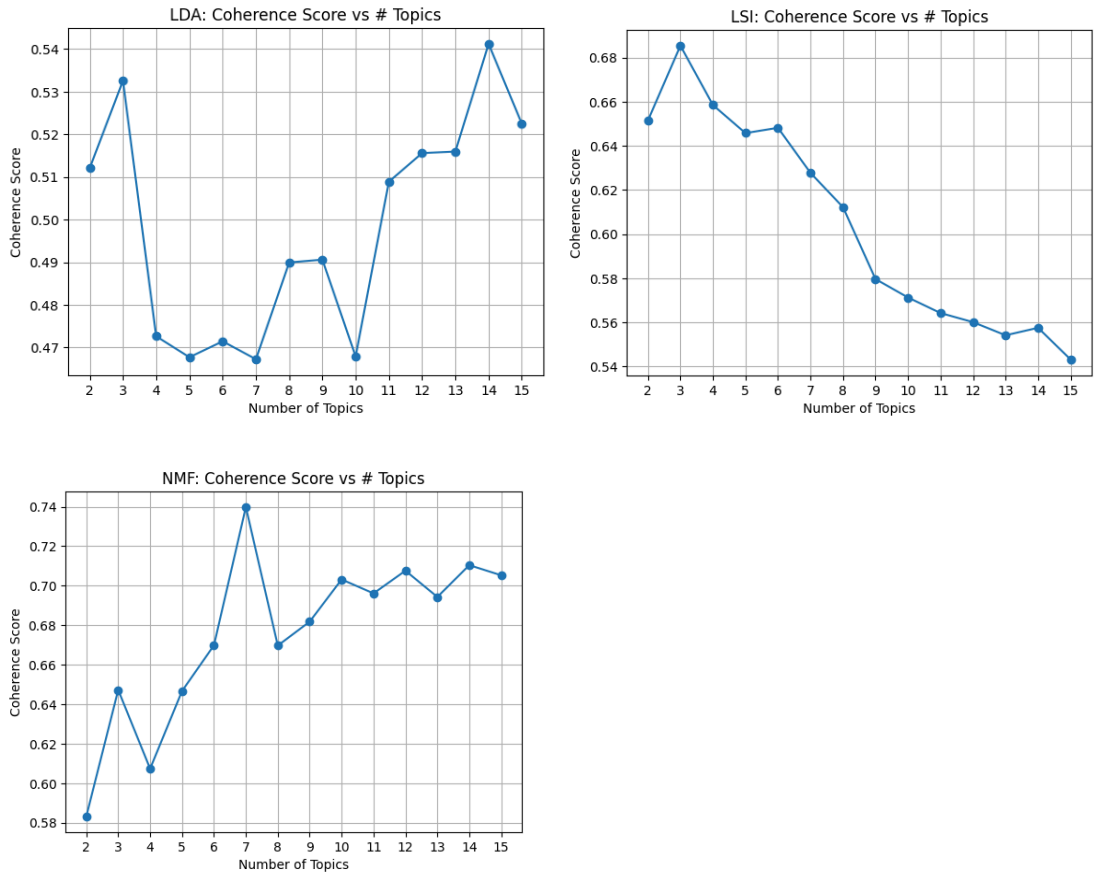

Figure 11: Three Coherence Score vs. Number of Topics Graphs for each model - LSI, LDA and NMF

Further, using Multi-dimensional scaling, the topic-term matrix is reduced to two dimensions and visualized in the form of a scatter plot below. This enables the observation of the geometric distance between distinct topics.

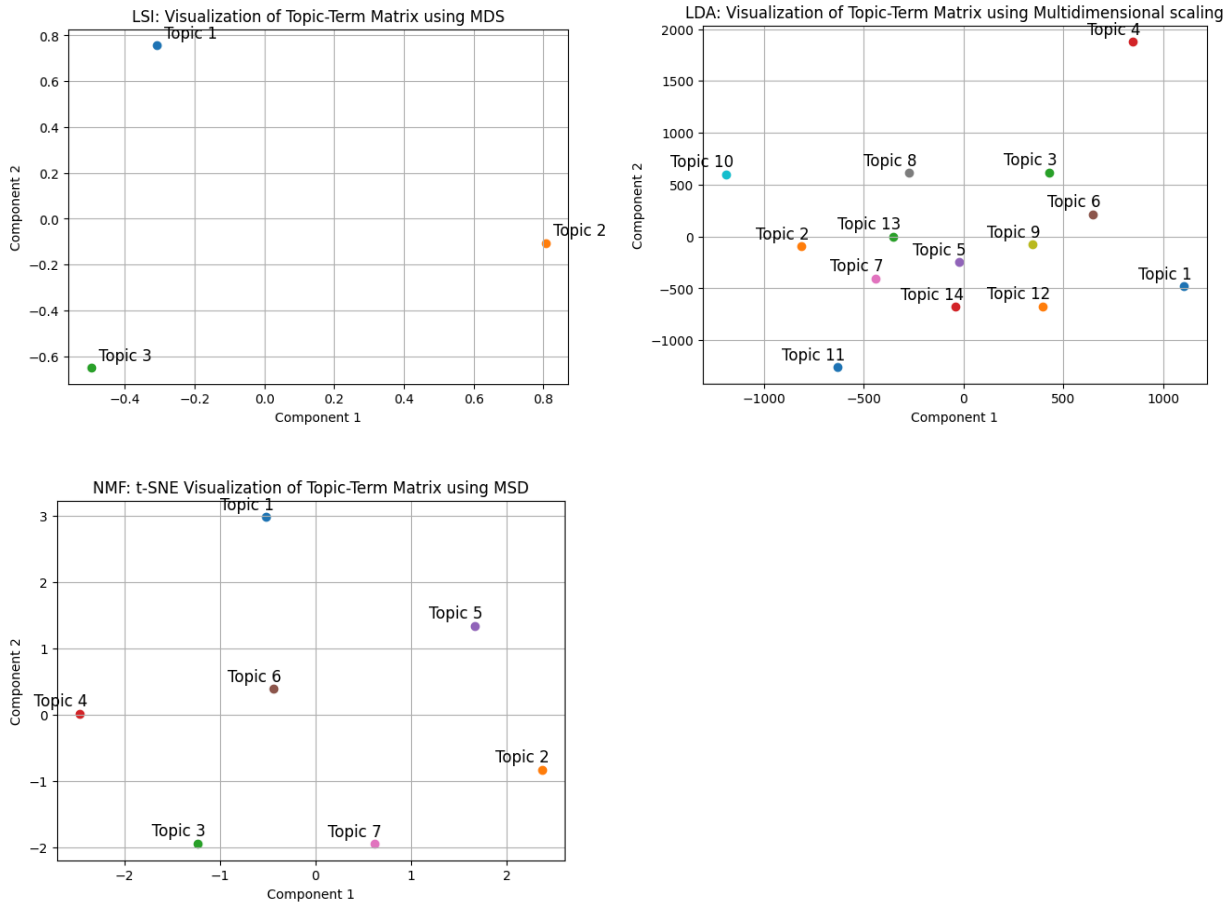

Figure 12: Three Topic-Term Matrices for each model - LSI, LDA, and NMF

By observing topics in their reduced form, it is evident that NMF presents topics that are evenly spaced out, whereas LDA exhibits clusters of some topics closely situated (very similar topics) with very few that are distinct enough.

Discussion

Supervised Learning

The supervised learning tasks of our project allowed us to implement and evaluate different types of supervised learning algorithms. One surprising result was the sentiment analysis logistic regression model achieving higher accuracy compared to the other algorithms we experimented with including Naive Bayes, Decision Trees, and Random Forests. With classification tasks, models like Naive Bayes or Decision Trees generally handle it better at capturing patterns in the data.

Surprisingly, the F1 scores using macro averaging for Conditional Random Fields and LSTM were quite similar. We had expected LSTM to have significantly better performance.

One of the challenges we encountered is having missing annotations for the NER task, so we generated our own. Named Entity Recognition labels were assigned to the training and test datasets using the Microsoft Phi2 language model. However, this lightweight model occasionally introduced mislabeled annotations, resulting in false positives and negatives. A more robust and accurate annotated dataset could have been obtained by leveraging advanced Large Language Models such as Llama-70B, GPT-4.0, etc.

Unsupervised Learning

NMF emerged as the most effective model for generating topics with optimal coherence, followed by LSI and LDA. Primarily TF-IDF and count vectorizer methods were utilized to represent documents in the vector space. Exploration of advanced techniques like word2vec and doc2vec could yield better performance. Additionally, the adoption of advanced topic evaluation methods such as perplexity and log likelihood could have provided valuable insights into the quality and coherence of the topics generated.

Another task we wanted to approach was text summarization, which was summarizing lengthy reviews into short sentences. Unfortunately, due to the restraints of time and resources, we did not get the chance to attempt it. Additionally, we could extend our solutions by developing a model to detect review bombing if we had more time. Unsupervised learning techniques, such as clustering or anomaly detection, could be explored and employed to identify reviews that show patterns that deviate from the general reviews which would indicate possible review bombing.

Ethical Considerations

Supervised Learning

One significant ethical consideration in the supervised learning portion of our project is the potential for sentiment bias during the sentiment analysis of restaurant reviews. Sentiment bias can occur if the model shows preference for detecting either positive or negative sentiments in reviews. This may potentially skew results of model performance. Additionally, cuisine can often be linked to different cultures so there must be careful consideration to ensure that the model does not favor certain types of restaurants or demographic groups over others.

Transparency is crucial in addressing this ethical concern. It's important to communicate how the sentiment analysis is done to provide context on how the model operates and it makes its decisions. Explaining the limitations and possible bias helps users understand the effectiveness of the model. The model's performance should be continually monitored along with feedback from users to limit bias and ensure fairness.

Unsupervised Learning

There are many ethical issues that need to be considered when applying topic modeling to restaurant reviews. It's important to be aware that identifying sensitive, negative, or controversial topics in restaurant reviews through topic modeling can potentially offend or be harmful to customers. This could lead to negative experiences for the customers. Uncovering damaging topics through topic modeling may also negatively affect restaurants' brand and image, which can lead to reputational harm. Restaurants can suffer negatively and lose their customers' trust. We need to be careful when communicating these topics. These issues can be addressed by providing context and disclaimers to ensure a balanced interpretation of the results and avoid misinterpretation of these sensitive topics. The results of the topic words can also be presented through aggregate reporting, which is where the topics are summarized and to prevent individual topics from hurting customers or the restaurant.

Another ethical concern is the model showing biases in identifying certain topics when applying topic modeling algorithms. Biased topic words can perpetuate stereotypes or misrepresent what the reviews actually meant, which can lead to skewing results of model performance. If we had more time, we could address these issues by developing a model to specifically detect biases.

Conclusion

While the predictions made through the models offer valuable guidance, it's important to emphasize that they should only be used as a guide rather than the ultimate decision-maker. The predictions should always be interpreted and vetted through ethical considerations and human judgment. We hope that our models can provide the insights restaurant businesses need to make informed decisions aimed at enhancing customer satisfaction for long-term success.

Statement of Work

Sangram Sanjiv Malekar	Minyan Gao	Maggie Oliver
Topic modeling, NER model development, Model evaluation, Report Writing for these tasks	Data preprocessing (cleaning), Sentiment Analysis Feature Importance, Report Writing for Introduction, Feature Engineering, Sentiment Analysis, Discussion, and Ethical Considerations	Data preprocessing (sampling dataset), Sentiment Analysis model development, Model evaluation, Failure Analysis, Report Writing for Sentiment Analysis Failure Analysis

References

Yelp Dataset Documentation. [Online]. Available: https://www.yelp.com/dataset/documentation/main

Kaggle Dataset: Restaurant Reviews. [Online]. Available: https://www.kaggle.com/datasets/joebeachcapital/restaurant-reviews

Kaggle Notebook: Sentiment Analysis of Restaurant Reviews. [Online]. Available: https://www.kaggle.com/code/apekshakom/sentiment-analysis-of-restaurant-reviews/notebook

GeeksforGeeks: Restaurant Review Analysis using NLP and SQLite. [Online]. Available: https://www.geeksforgeeks.org/restaurant-review-analysis-using-nlp-and-sqlite/

GitHub Repository: Topic Modelling on BBC News Article. [Online]. Available: https://github.com/Malu2203/Topic-modelling-on-BBC-news-article

Greene, D. (n.d.). Topic Modelling with Scikit-Learn. [Online]. Available: http://derekgreene.com/slides/topic-modelling-with-scikitlearn.pdf

SIADS 543 Notebooks: Advanced Machine Learning. [Online]. Available: https://www.coursera.org/learn/siads543/

FreeCodeCamp. (n.d.). Advanced Topic Modeling: How to Use SVD & NMF in Python. [Online]. Available: https://www.freecodecamp.org/news/advanced-topic-modeling-how-to-use-svd-nmf-in-python/

SIADS 642 Notebooks: Deep Learning. [Online]. Available: https://www.coursera.org/learn/siads642/

Zhou, B. (n.d.). Named Entity Recognition (NER) Using Keras LSTM and Spacy. [Online]. Available: https://zhoubeiqi.medium.com/named-entity-recognition-ner-using-keras-lstm-spacy-da3ea63d24c5

WildML. (2015, September). Recurrent Neural Networks Tutorial - Part 1: Introduction to RNNs. [Online]. Available: https://web.archive.org/web/20211110115049/http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

PyTorch Tabular Multiclass Classification. [Online]. Available: https://towardsdatascience.com/pytorch-tabular-multiclass-classification-9f8211a123ab

Medium - Analytics Vidhya. (n.d.). Deep Learning Basics: Weight Decay. [Online]. Available: https://medium.com/analytics-vidhya/deep-learning-basics-weight-decay-3c68eb4344e9

Python CRFSuite Documentation. [Online]. Available: https://python-crfsuite.readthedocs.io/en/latest/

Medium - ML2Vec. (n.d.). Overview of Conditional Random Fields. [Online]. Available: https://medium.com/ml2vec/overview-of-conditional-random-fields-68a2a20fa541

Towards Data Science. (n.d.). Conditional Random Field Tutorial in PyTorch. [Online]. Available: https://towardsdatascience.com/conditional-random-field-tutorial-in-pytorch-ca0d04499463

Feregrino, F. (n.d.). Conditional Random Fields in Python: Sequence Labelling (Part 4). [Online]. Available: https://dev.to/fferegrino/conditional-random-fields-in-python-sequence-labelling-part-4-5ei2

NLTK Project. (n.d.). nltk.classify.scikitlearn module. [Online]. Available: https://www.nltk.org/api/nltk.classify.scikitlearn.html

Appendix A: Dataset Catalog

All read-only copies of our datasets and notebooks can be accessed through the Google Drive link: https://drive.google.com/drive/folders/14XiECiI5p57SKoTiCZCuKD1TK-QNGRuq?usp=sharing

Dataset	Dataset Name	Dataset Description
Yelp Dataset	yelp_academic_dataset_business.json	The Yelp dataset with businesses and their locations, which is used to help sample our data.
	yelp_academic_dataset_review.json	The Yelp dataset with review text. Note: This dataset size is 5GB and it couldn't be uploaded so please extract it us
NER	Yelp (folder)	Several datasets with annotations from the Yelp dataset for the NER model
	Kaggle (folder)	The Kaggle external dataset and several smaller datasets with annotations from the Kaggle dataset for the NER model.
	Model (folder)	Intermediate models created by NER model training phase and used for inference.

Appendix B: Notebook Catalog

All read-only copies of our datasets and notebooks can be accessed through the Google Drive link: https://drive.google.com/drive/folders/14XiECiI5p57SKoTiCZCuKD1TK-QNGRuq?usp=sharing

Notebook Category	Notebook Name	Notebook Description
Data Preprocessing	data_prep_restaurant_reviews.ipynb	Loads, cleans and preprocesses our data to the clean sample of Philadelphia data we want.
Supervised Learning	sentiment_classification_models.ipynb	Sentiment analysis model to sort reviews into star categories. Includes Feature Importance and Failure Analysis.
	synthetic_annotation_LLM_MSFT_Phi2.ipynb	LLM based synthetic annotation on review dataset
	NER_CRF.ipynb	CRF Model development for NER
	NER_RNN.ipynb	LSTM Model development for NER
	NER_RNN_HyperTune.ipynb	LSTM Model Fine Tuning
	NER_Inference.ipynb	To run NER inference using a fine tuned model. Includes Failure analysis.
Unsupervised Learning	Topic_Model.ipynb	Topic modeling using LDA, LSI, NMF