MSc in Business Analytics

Machine Learning and Content Analysis

Professor: Papageorgiou Haris

Assistant Professor: Perakis Georgios

NLP Sentiment Analysis in Hotel Bookings Reviews

Assignment, Report

September 2024, Athens

ARGYROPOULOS DIMOSTHENIS (f2822317)

PAPAGEORGIOU THEODOROS(f2822310)

Contents

Introduction

In the hospitality industry, sentiment analysis plays a crucial role in shaping business strategies. As customer feedback becomes increasingly accessible through online platforms, hotels must effectively analyze this information to measure customer satisfaction and identify areas for improvement. By understanding the sentiments expressed in reviews, hotel operators can make informed decisions to enhance guest experiences, tailor marketing efforts, and ultimately increase customer loyalty.

Our company recognizes the profound impact that customer reviews have on business performance in the hospitality industry. By systematically analyzing the sentiments expressed in these reviews, we aim to uncover actionable insights that can drive strategic decision-making. We hypothesize that a well-developed sentiment analysis model will not only enhance our understanding of guest experiences but also enable us to identify specific aspects of service that resonate with customers. This data-driven approach will allow us to construct a comprehensive strategic plan that prioritizes guest satisfaction, optimizes service delivery, and ultimately fosters a more competitive edge in the market.

As a team of two members at this Hotel Company, driven by our strong dedication to generating new ideas and ensuring customer satisfaction, we're prepared to overcome

these challenges. We're excited to bring in a solution that could really change things. Our idea is NLP Sentiment Analysis which focuses on categorizing reviews based on the sentiment of customers.

Project Description

The goal of this project is to develop a robust Natural Language Processing (NLP) sentiment analysis model tailored specifically for hotel reviews. This model will categorize customer feedback into three sentiment classes: positive, neutral, and negative. To address the challenges associated with an imbalanced dataset and the actual meaning of human language, we will employ advanced NLP techniques to capture the contextual meaning of words in reviews.

Our solution based on training and validating our model on a diverse dataset of hotel reviews, and we aim to provide hotel operators with a reliable tool for understanding guest sentiments and improving service quality. The insights derived from this analysis will guide strategic initiatives, enabling hotels to enhance customer experiences, effectively address concerns, and ultimately drive loyalty and revenue growth. Our project is not only an academic endeavor but also a practical solution aimed at transforming the way hotels leverage customer feedback for continuous improvement.

Our Vision

Our vision is to change the hospitality industry by using sentiment analysis to turn customer feedback into useful insights. We want to help hotel operators understand and respond to what guests are saying, creating a culture that values improvement and great service. By using advanced language processing technologies in their daily operations, hotels can better connect with their guests and enhance their experiences. Ultimately,

we envision a future where every hotel can use customer insights to make smart decisions, leading to better service and long-term growth in a competitive market.

Data Collection

For this sentiment analysis project, we used a dataset from data.world (Datafiniti, Hotel Reviews) that contains hotel reviews collected from different online sources. This dataset includes a variety of guest experiences, making it perfect for our analysis. By using this collection of customer feedback, we can understand different feelings about hotel services. The dataset has important information, such as the review text and star ratings, which will help us train and test our sentiment analysis model effectively. Choosing the right data is essential to ensure our results truly represent what guests think and feel about hotels. The dataset used can be found in https://data.world/datafiniti/hotel-reviews.

Dataset Overview/Processing

The dataset we used for our sentiment analysis project consists of 10,000 rows and 26 columns, but for our analysis, we focused primarily on two key features: the reviews themselves and their corresponding star ratings. The relevant columns in the dataset are:

- **reviews.text:** This column contains the actual text of the customer reviews, which provides insights into guests' experiences and sentiments about the hotels.
- **reviews.rating:** This column includes the star rating given by the customers, which serves as a numerical representation of their overall satisfaction.

By concentrating on these two columns, we can effectively analyze the relationship between the text of the reviews and the star ratings, allowing us to build a sentiment analysis model that accurately captures guest sentiments in the hospitality industry.

Additionally, we removed 1 observation with a missing value and 18 duplicate observations from the dataset to ensure the integrity of our analysis, so the final dataset consists of 9,981 rows.

Dataset Descriptive Statistics for Numeric column (reviews.rating):

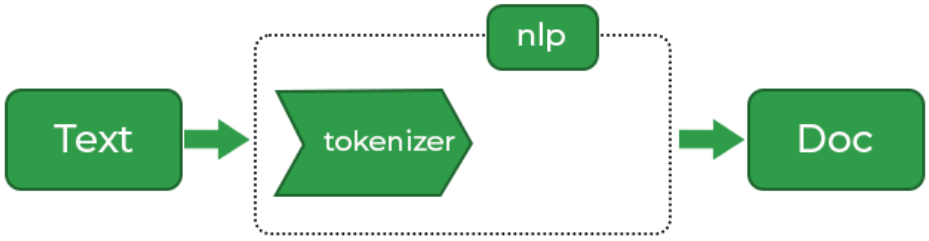
count	mean	std	min	25%	50%	75%	max
10000.0	4.03	1.16	1.0	3.35	4.0	5.0	5.0

Interpretation: The summary statistics for the “reviews.rating” column reveal a generally posotive perception among reviewers. With a mean rating of 4.03, the average review suggests that customers are satisfied with their experiences. The standard deviation of 1.16 indicates a moderate variability in ratings, showing that while many customers rate their experiences highly, some ratings vary significantly. The minimum rating of 1.00 highlights the presence of dissatisfied customers, while the maximum rating of 5.00 indicates that some reviewers were exceptionally pleased. The interquartile range, with 25% of ratings below 3.35 and 75% below 5.00, suggests that the majority of reviews cluster towards the higher end of the rating scale, reflecting overall positive sentiments.

Methodology

In this project, several models were developed and evaluated to perform sentiment analysis on hotel reviews. The first step involved preprocessing the text data to ensure its suitability for analysis. The reviews column was cleaned by addressing issues such as excess spaces and converting all letters to lowercase, thereby standardizing the text.

Following this, the cleaned text was tokenized, breaking down the reviews into individual words while retaining meaningful context. This tokenization process allowed for a more nuanced representation of the data, facilitating the subsequent embedding creation.



For the word embeddings, we utilized **Gensim** to generate **Word2Vec** embeddings, capturing the semantic relationships between words based on their context within the corpus. To prepare the data for **Logistic Regression** and **FastText** models, we combined the tokenized words into single sentences, ensuring that the models could interpret the input correctly. This was achieved by creating a new column, cleaned_text_joined, which contained the joined tokens.

After embedding generation, we implemented various **Natural Language Processing (NLP)** techniques using multiple embedding methods. For each embedding method, three architectures of a **Multi-Layer Perceptron (MLP)** were constructed:

1. A **Simple MLP** architecture with one hidden layer consisting of 128 neurons.
2. An MLP with three hidden layers, which enhanced the model's complexity to better capture intricate patterns in the data.
3. An MLP with four hidden layers, further increasing the model's capacity for representation.

Additionally, a **self-attention model** was developed using the **BERT tokenizer**, leveraging the power of transformer architecture to obtain contextual embeddings that effectively capture the nuances in textual data.

The performance of all models was assessed using a comprehensive set of evaluation metrics, including **accuracy**, **F1 score**, **recall**, **precision**, **confusion matrix**, **information loss**, and **validation loss**. This multi-faceted evaluation approach enabled a thorough comparison of model performance, leading to the identification of the most

effective architecture for sentiment classification, while ensuring that the chosen model not only accurately predicts sentiment but also generalizes well to unseen data.

Let's explain the architectures and dimensions:

Each model was designed to find patterns in the sentiment of hotel reviews while balancing complexity and performance. The **Simple MLP** architecture, with one hidden layer of 128 neurons, served as a starting point to establish a baseline performance that was efficient in terms of computation.

The second architecture had **three hidden layers** with different sizes: the first layer had 256 neurons, the second had 128 neurons, and the third had 64 neurons. This design allowed the model to learn more complex features from the data. Each layer gets smaller, which is a common practice in deep learning, as the later layers focus on refining the features learned from the earlier ones.

The most complex model included **four hidden layers**. It had the same first three layers as the three-layer model (256, 128, and 64 neurons) but added an extra hidden layer to learn even more detailed features. This structure aimed to improve the model's ability to tell apart subtle differences in sentiment within the reviews.

For the **self-attention model using the BERT tokenizer**, the design was different from the MLP models. It used attention mechanisms to evaluate which words in a review were most important. This approach helps the model understand the context of the text better, making it a good fit for analyzing sentiments.

By comparing these different models, we aimed to assess not only how well they performed but also how well they could adapt to the imbalanced classes in the dataset. The variations in depth and complexity across the models helped us understand how design choices affect performance, guiding us to select the best model for sentiment classification.

Model Results/Evaluation/Error

This section presents the performance of the models developed for sentiment analysis using various evaluation metrics, including **accuracy**, **F1 score**, **recall**, **precision** and **validation loss**. Since the dataset is imbalanced, F1 score becomes one of the most important metrics to consider, as it balances precision and recall. Each model was evaluated on the same training and test sets by splitting the original dataset into 80% for training and 20% for testing, using a consistent random state across all models to ensure comparability. The performance metrics provided insights into each model's ability to generalize and accurately classify sentiments.

1. Logistic Regression

Logistic Regression was the baseline model. It achieved:

- **Accuracy**: 82.32%
- **F1 score**: 0.7876
- **Precision**: 0.7948
- **Recall**: 0.8232

This model provided a simple starting point, but as expected, its performance was limited compared to more complex models.

2. Word2Vec Models

- **Simple MLP**: This model with one hidden layer of 128 neurons achieved:
    - **Accuracy**: 78.97%
    - **F1 score**: 0.7334
    - **Precision**: 0.7383
    - **Recall**: 0.7897
    - **Validation Loss**: 0.5420
- **3 Hidden Layers** (256, 128, 64 neurons): This deeper model showed improved performance, with:
    - **Accuracy**: 79.32%
    - **F1 score**: 0.7433

    - **Precision**: 0.7391
    - **Recall**: 0.7932
    - **Validation Loss**: 0.5474
- **4 Hidden Layers**: The most complex Word2Vec model, adding an additional hidden layer, yielded:
    - **Accuracy**:79.32%
    - **F1 score**: 0.7328
    - **Precision**: 0.7351
    - **Recall**: 0.7932
    - **Validation Loss**: 0.5416

3. FastText Models

Like the Word2Vec models, three architectures were tested using FastText embeddings:

- **Simple MLP** (1 hidden layer):
    - **Accuracy**: 79.87%
    - **F1 score**: 0.7673
    - **Precision**: 0.7536
    - **Recall**: 0.7987
    - **Validation Loss**: 0.5016
- **3 Hidden Layers**:
    - **Accuracy**: 80.92%
    - **F1 score**: 0.7734
    - **Precision**: 0.7679
    - **Recall**: 0.8092
    - **Validation Loss**: 0.4897
- **4 Hidden Layers**:

- **Accuracy**:80.72%
- **F1 score**: 0.7745
- **Precision**: 0.7668
- **Recall**: 0.8072
- **Validation Loss**: 0.4997

4. TF-IDF Models

TF-IDF embeddings were also used in the same architectures:

- **Simple MLP** (1 hidden layer):
  - **Accuracy**: 79.32%
  - **F1 score**: 0.7750
  - **Precision**: 0.7661
  - **Recall**: 0.7932
  - **Validation Loss**: 0.7601
- **3 Hidden Layers**:
  - **Accuracy**: 79.32%
  - **F1 score**: 0.7823
  - **Precision**: 0.7758
  - **Recall**: 0.7932
  - **Validation Loss**: 0.8526
- **4 Hidden Layers**:
  - **Accuracy**: 78.32%
  - **F1 score**: 0.7809
  - **Precision**: 0.7923
  - **Recall**: 0.7832

- **Validation Loss**: 0.7803

5. Self-Attention Model (BERT Tokenizer)

The self-attention model using the BERT tokenizer performed as follows:

- **Accuracy**: 85.03%
- **F1 score**: 0.8499
- **Precision**: 0.8501
- **Recall**: 0.8503
- **Validation Loss**: 0.5978

Summary of Results:

- Logistic regression served as a simple baseline. Its accuracy is decent, but both the F1 score and precision are lower compared to more advanced models. As expected, it's outperformed by more complex architectures.
- The Word2Vec models performed worse than the baseline in accuracy and F1 score, and the validation loss doesn't improve significantly across architectures. These models don't seem to offer significant advantages over logistic regression.
- The FastText models outperform Word2Vec in accuracy and F1 score across all architectures. The 3 Hidden Layers FastText model gives the best validation loss (0.4897) and shows strong performance in accuracy (80.92%) and F1 score (0.7734), making it a more reliable model compared to Word2Vec and logistic regression.
- The TF-IDF models showed reasonable F1 scores, but their validation losses are much higher compared to FastText and Word2Vec models. The best F1 score from these models (0.7823) is decent, but the high validation loss (0.8526) indicates that TF-IDF might not generalize as well.
- The self-attention model using BERT is the top-performing model. It has the highest accuracy (85.03%) and the best F1 score (0.8499). Although its validation loss (0.5978) is slightly higher than FastText models, the model

excels in all other performance metrics. BERT's ability to capture context makes it a very strong model for this task.

Conclusion: Best Model

The self-attention model (BERT) stands out as the best model, based on its superior accuracy and F1 score. It balances precision and recall well and outperforms all other models significantly. Although the FastText (3 hidden layers) model has a better validation loss, the overall performance of BERT makes it the most effective choice for sentiment classification in this project.

ROC curve:

The Receiver Operating Characteristic (ROC) curve is an essential tool for evaluating the performance of our BERT model across different classes. By plotting the true positive rate against the false positive rate, we can visualize how effectively the model distinguishes between the various sentiment categories. In this analysis, we have calculated the Area Under the Curve (AUC) for each class, providing a quantitative measure of the model's discriminative ability. The following interpretations highlight the model's performance in classifying each sentiment category.

- **Class 0 (Negative Sentiment) - AUC = 0.97:**

An AUC of 0.97 indicates that the model performs exceptionally well in identifying negative sentiments. This high AUC value suggests that the model has a strong ability to correctly classify negative reviews, achieving a high true positive rate while maintaining a low false positive rate. In practical terms, this means that the model is highly reliable in recognizing when a review expresses negative sentiments.

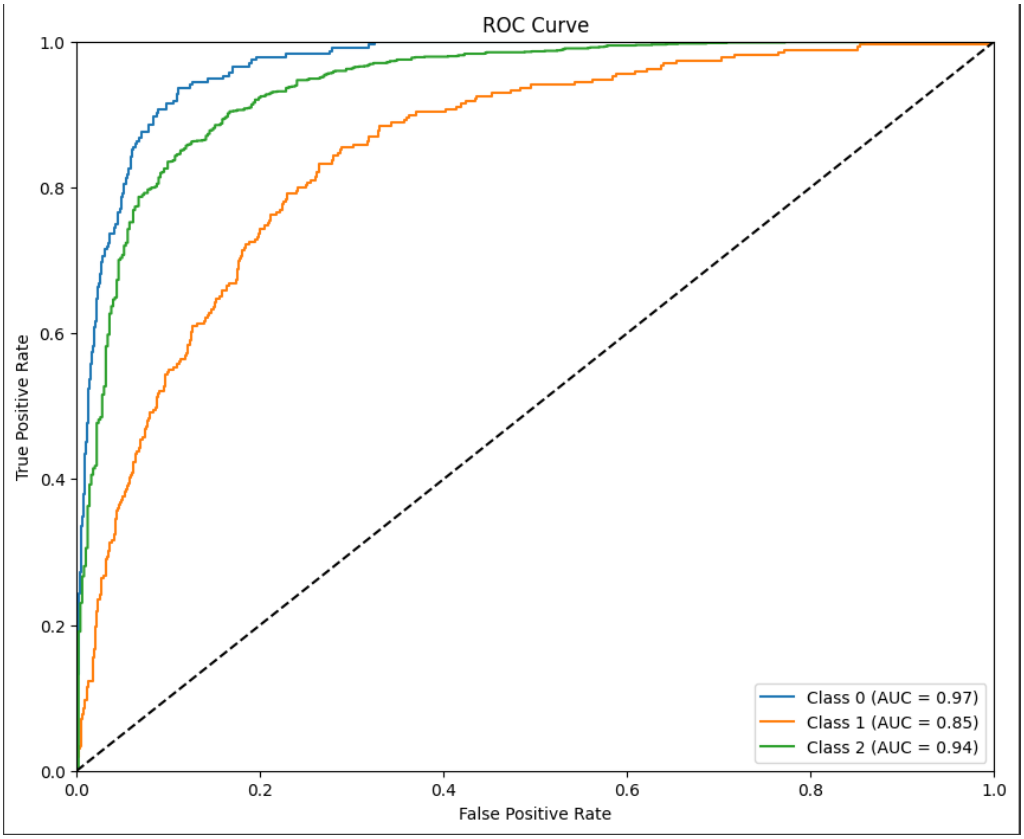
- **Class 2 (Positive Sentiment) - AUC = 0.94:**

The AUC of 0.94 for positive sentiment indicates robust performance as well. This value signifies that the model effectively distinguishes positive reviews from other classes. While slightly lower than the negative sentiment AUC, it still reflects a high

level of accuracy, showing that the model can confidently classify positive sentiments with minimal misclassification.

- **Class 1 (Neutral Sentiment) - AUC = 0.85:**

The AUC of 0.85 for neutral sentiment, while still considered good, is lower compared to the other two classes. This suggests that the model has a moderate ability to differentiate neutral reviews from positive and negative ones. The lower AUC indicates that there may be some overlap in features between neutral and the other sentiments, which could result in a higher rate of misclassification for neutral reviews.



Libraries and Tools

In this project, we employed a range of powerful tools and libraries to develop and evaluate our sentiment analysis models effectively. The primary development environment was Google Colab, which facilitated collaboration and provided robust

GPU support for efficient model training. We wrote our code in Python, leveraging libraries such as NumPy and Pandas for data manipulation and analysis. Matplotlib and Seaborn were used for visualizing data distributions and model performance, enhancing our understanding of the results.

For natural language processing, we utilized NLTK for tokenization and stopword removal, and Gensim to create Word2Vec and FastText embeddings, capturing the semantic relationships within the text data. The implementation of our machine learning models was supported by Scikit-learn, which provided tools for model evaluation, feature extraction with TfidfVectorizer, and training logistic regression models.

To build and fine-tune our self-attention model, we leveraged the Transformers library from Hugging Face, which allowed us to work with BERT model along with its tokenizer.

We also incorporated PyTorch for model training and optimization, utilizing its robust framework for creating neural networks. The combination of these libraries and tools provided a solid foundation for tackling the challenges of sentiment analysis on our dataset.

Discussion - Future Work

For future work, several enhancements could be made to improve the sentiment analysis models and their practical applications. One potential avenue is to implement a recommendation system that provides tailored solutions based on the sentiment expressed in the reviews. For instance, if a review indicates dissatisfaction with a particular aspect of a hotel, the system could suggest specific improvements or amenities that would enhance the guest experience. Additionally, expanding the models to support multiple languages would significantly broaden their applicability, allowing businesses to analyze sentiment in diverse markets and better understand customer feedback from various linguistic backgrounds. Furthermore, incorporating more advanced techniques such as transfer learning or ensemble methods could lead to improved model performance and robustness, particularly in dealing with imbalanced datasets. Finally, continuous model fine-tuning and evaluation based on real-world

feedback could help adapt the system to changing customer sentiments and preferences over time.

Members and Roles

Our team consists of two university students who worked together on the sentiment analysis project. We stayed closely connected throughout the entire process, having regular discussions to make sure we were on the same page. Instead of splitting up tasks, we decided to collaborate on every part of the project. This included preparing the data, building the models, and evaluating their performance. By exploring the dataset and interpreting the results together, we were able to share our knowledge and learn from each other. This teamwork helped us better understand sentiment analysis and led to a successful project outcome.

References/Bibliography

- Zhang, L., & Liu, B. (2017). Sentiment Analysis and Opinion Mining. Encyclopedia of Machine Learning and Data Mining, 1152–1161. doi:10.1007/978-1-4899-7687-1_907
- Jain, K., & Kaushal, S. (2018). A Comparative Study of Machine Learning and Deep Learning Techniques for Sentiment Analysis. 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO). doi:10.1109/icrito.2018.8748793
- pytorch_with_example
- Course slides

END OF REPORT