Sentiment Analysis of Amazon Food Reviews

**Ahmed Awad Sayed∗ Mohamed Aref Attia† Nuha Nasser ‡**

**Fares Mohamed Abdullah§ Ahmed Raafat ahmed ¶**

October 14, 2024

# Abstract

This project delves into the domain of sentiment analysis, specifically focusing on analyzing the sentiment behind Amazon Fine Food reviews. By utilizing various machine learning techniques, the goal is to classify user-generated reviews into distinct sentiment categories, namely positive, neutral, and negative, based solely on the textual content of the reviews. The dataset comprises user-written text reviews along with corresponding ratings, serving as a basis for model training and validation. For this purpose, a hybrid deep learning model combining a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network was implemented. The performance of the model was rigorously evaluated using several standard metrics, including accuracy, precision, recall, and F1 score, among others, to measure its effectiveness in classifying sentiments accurately.

# Introduction

Sentiment analysis, often referred to as opinion mining, is a subfield of natural language processing (NLP) that aims to interpret and categorize emotions within textual data. In the context of consumer reviews, sentiment analysis enables businesses and individuals to better understand customer opinions, helping to inform decisions related to products, marketing strategies, and overall consumer satisfaction.

This project addresses sentiment analysis in the context of Amazon Fine Food reviews, an area that has seen increasing importance due to the growing reliance on online reviews for purchasing decisions. The primary aim of this project is to analyze these reviews, classify them into one of three sentiment categories—positive, neutral, or negative—and thereby offer insights into customer satisfaction. The dataset utilized consists of thousands of text reviews, each associated with a numerical rating, which serves as a proxy for the sentiment expressed in the review.

Natural language processing techniques such as tokenization, removal of stopwords, and lemmatization were applied to prepare the text for model training. The machine learning model was designed to interpret the processed text and predict the sentiment class. The CNN-LSTM model chosen for this project was selected for its ability to extract both spatial and sequential features from the text, allowing for a more nuanced understanding of sentiment compared to simpler models.

# Dataset and Preprocessing

The dataset used in this project is derived from Amazon Fine Food reviews, a comprehensive collection of customer feedback on a wide variety of food products available on the platform. Each review consists of multiple features, including the text of the review itself, the product ID, the user’s profile name, and a numerical rating between 1 and 5. For the purposes of this analysis, we focused on two main features: the text of the review and the associated rating.

## Initial Exploration and Data Cleaning

Upon importing the dataset, an initial exploration was conducted to understand its structure and contents. The dataset contained approximately 500,000 reviews, with the majority of reviews falling into the ‘positive‘ sentiment category (ratings 4 and 5). This imbalance in the distribution of sentiment categories necessitated careful handling to prevent the model from becoming biased toward predicting the majority class.

The text reviews were then preprocessed through a series of steps designed to clean and standardize the data for further analysis. These steps included: • \*\*Converting text to lowercase:\*\* To ensure uniformity across the dataset, all text was converted to lowercase. This step eliminates discrepancies between words that may be capitalized in some reviews but not in others. • \*\*Removing punctuation:\*\* Punctuation marks, while useful in human communication, add noise to the data in machine learning models. As a result, all punctuation marks were removed from the reviews. • \*\*Tokenization:\*\* Each review was split into individual tokens (i.e., words or subwords) to facilitate further processing. This allows the model to treat each token as a unit of meaning.

• \*\*Removing stopwords:\*\* Common words that do not carry significant meaning, such as ”and,” ”the,” and ”is,” were removed to focus on the words that contribute most to the sentiment expressed in the reviews. • \*\*Lemmatization:\*\* Lemmatization was performed to reduce words to their base or root forms. For example, the word ”running” would be reduced to ”run,” helping to group together different forms of the same word.

After completing these preprocessing steps, the dataset was examined for any missing values or outliers. Missing values were either imputed or removed, depending on the context, and outliers were handled appropriately to ensure the dataset was clean and ready for model training.

## Handling Imbalanced Data

One of the challenges in sentiment analysis is dealing with imbalanced data, where certain classes (e.g., positive reviews) are overrepresented compared to others (e.g., negative reviews). In this dataset, positive reviews were significantly more frequent than neutral or negative reviews, which could lead to biased model predictions.

To address this issue, class balancing techniques such as downsampling and oversampling were explored. Downsampling was ultimately chosen as the preferred method for this project, which involved randomly reducing the number of samples in the positive class to match the number of samples in the neutral and negative classes. This ensured that each sentiment class was represented equally, improving the model’s ability to generalize across all classes.

## Mapping Sentiments to Ratings

The Amazon Fine Food reviews dataset contains numerical ratings ranging from 1 to 5, with 1 indicating extreme dissatisfaction and 5 indicating extreme satisfaction. For the purposes of sentiment analysis, these ratings were mapped to three distinct sentiment categories as follows:

* \*\*Negative sentiment:\*\* Reviews with ratings of 1 and 2 were classified as expressing negative sentiment. These reviews typically contained complaints or dissatisfaction with the product.
* \*\*Neutral sentiment:\*\* Reviews with a rating of 3 were classified as neutral. These reviews often contained balanced opinions, neither strongly positive nor strongly negative.
* \*\*Positive sentiment:\*\* Reviews with ratings of 4 and 5 were classified as positive, indicating a high level of satisfaction with the product.

This mapping allowed the model to focus on predicting sentiment rather than attempting to predict the exact numerical rating, simplifying the classification task while maintaining the richness of the original data.

# Model Architecture

For the task of sentiment classification, a hybrid model combining a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network was employed. This combination was chosen because it allows the model to capture both the local relationships between words (via the CNN) and the sequential dependencies across the entire review (via the LSTM). The resulting model is well-suited for tasks that require both spatial and temporal understanding of the data, making it ideal for sentiment analysis of text.

## CNN-LSTM Model Design

The CNN-LSTM model was designed with several key components, each playing a crucial role in extracting and processing the information from the text: • \*\*Embedding layer:\*\* The first layer of the model is an embedding layer, which converts the input text into dense vector representations. This allows the model to capture semantic similarities between words, where words with similar meanings are represented by similar vectors. The embedding layer was configured with a vocabulary size of 10,000 (to cover the most frequently occurring words in the dataset) and an embedding dimension of 128.

* \*\*Convolutional layer (Conv1D):\*\* Following the embedding layer, a 1D convolutional layer was applied to extract local patterns from the text. This layer uses 128 filters and a kernel size of 5, enabling it to detect word n-grams and short phrases that may indicate sentiment.
* \*\*MaxPooling layer:\*\* A max-pooling layer was added after the convolutional layer to reduce the dimensionality of the feature maps. This layer takes the maximum value from each feature map over a pool size of 2, preserving the most important features while reducing the computational load for subsequent layers.
* \*\*LSTM layer:\*\* After the convolutional and pooling layers, an LSTM layer with 128 units was applied. The LSTM layer is responsible for capturing the sequential dependencies in the text, such as the relationship between words that appear far apart in the review. This helps the model understand the overall sentiment expressed in the review, rather than just focusing on individual words or phrases.
* \*\*Dense layer:\*\* A fully connected dense layer with 128 units and ReLU activation was used to introduce non-linearity into the model and allow for more complex representations of the data.
* \*\*Dropout layer:\*\* To prevent overfitting, a dropout layer with a rate of 0.5 was added. This layer randomly sets half of the units to zero during training, forcing the model to learn more robust features that generalize better to unseen data.
* \*\*Output layer:\*\* The final layer of the model is a softmax output layer with 3 units, corresponding to the three sentiment classes (positive, neutral, and negative). The softmax activation function was used to convert the output into probability distributions, allowing the model to predict the most likely sentiment class for each review.

The model was trained using the categorical cross-entropy loss function, which is commonly used for multi-class classification tasks. The Adam optimizer was selected for its ability to adapt the learning rate during training, helping the model converge faster and achieve better results. The model was trained for 10 epochs, with the performance monitored on a validation set to prevent overfitting.

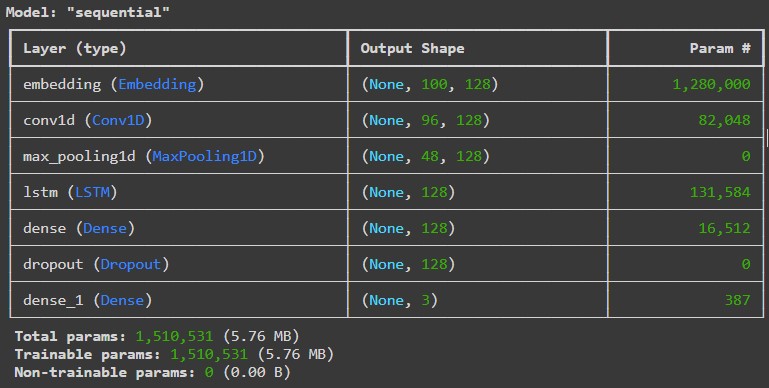


Figure 1: CNN-LSTM Model Architecture

# Evaluation

Once the model was trained, its performance was evaluated on a test set that had not been seen during the training process. Several evaluation metrics were used to assess the model’s performance, providing a comprehensive understanding of its ability to classify sentiment accurately.

## Test Set Evaluation

The model achieved an overall accuracy of 84.37% on the test set, meaning that it correctly predicted the sentiment class for approximately 84 out of every 100 reviews. However, accuracy alone does not provide a complete picture of the model’s performance, as it can be influenced by the class distribution. Therefore, additional metrics such as precision, recall, and F1 score were calculated for each sentiment class to provide a more detailed evaluation.

Table **??** shows the precision, recall, and F1 score for each sentiment class. Precision measures the proportion of correct positive predictions, recall measures the proportion of actual positives that were correctly identified, and F1 score is the harmonic mean of precision and recall, providing a balanced measure of the model’s performance.

Table 1: Performance Metrics for Each Sentiment Class

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Precision | Recall | F1 Score |
| Negative | 0.81 | 0.82 | 0.82 |
| Neutral | 0.74 | 0.70 | 0.72 |
| Positive | 0.89 | 0.92 | 0.91 |

## Confusion Matrix

To further analyze the model’s predictions, a confusion matrix was generated (Figure **??**). The confusion matrix provides a breakdown of how many reviews from each sentiment class were correctly or incorrectly classified. This helps to identify any systematic errors made by the model, such as whether it tends to confuse neutral reviews with either positive or negative reviews.

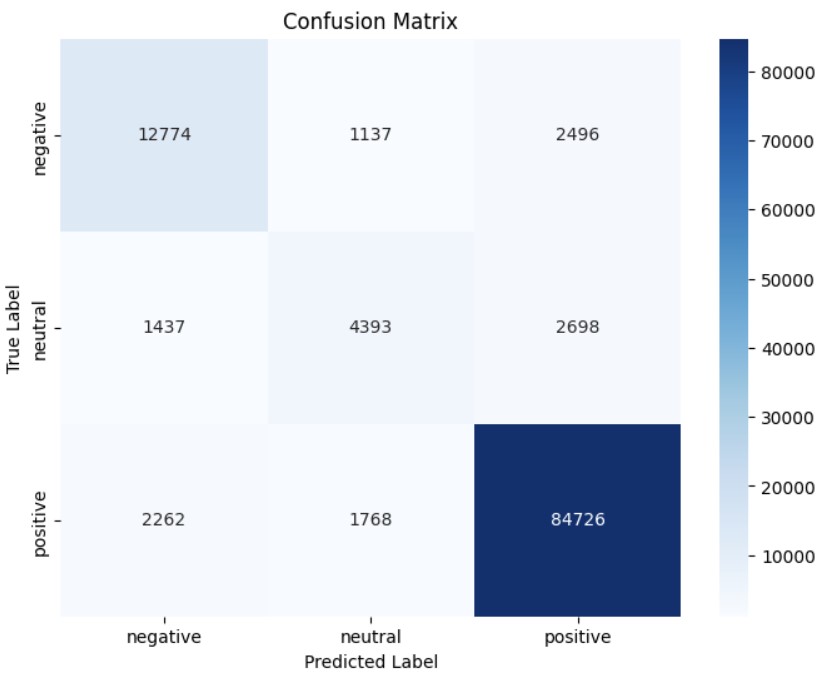


Figure 2: Confusion Matrix

## Precision-Recall Curve

A precision-recall curve was also plotted for each sentiment class to illustrate the trade-off between precision and recall (Figure **??**). This curve is particularly useful in cases where the class distribution is imbalanced, as it shows how well the model performs across different thresholds for predicting each class. By analyzing the precision-recall curve, it is possible to adjust the model’s threshold to achieve the desired balance between precision and recall for each sentiment class.

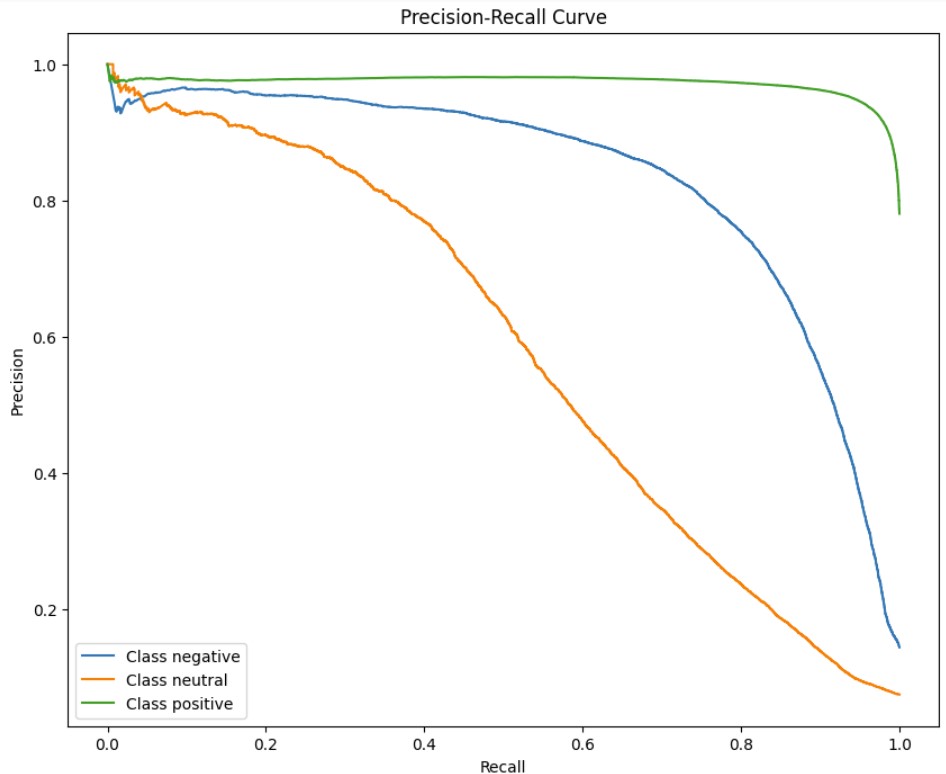


Figure 3: Precision-Recall Curve

# Conclusion

In conclusion, this project successfully implemented a sentiment analysis model for classifying Amazon Fine Food reviews into positive, neutral, and negative sentiment categories. The CNN-LSTM hybrid model demonstrated strong performance, achieving an accuracy of over 84% and showing balanced precision and recall across all sentiment classes. The project highlights the importance of thorough data preprocessing, as well as the effectiveness of combining convolutional and recurrent neural networks for sentiment classification tasks.

While the results are promising, there are several avenues for future improvement. For example, experimenting with alternative model architectures, such as transformers or attention-based models, could lead to further performance gains. Additionally, incorporating additional features, such as the length of the review or metadata about the product, could provide the model with more context and improve its ability to predict sentiment accurately.

In summary, this project demonstrates the potential of deep learning models in the field of natural language processing and provides a foundation for further exploration into more advanced sentiment analysis techniques.