Amazon Reviews: Text Classification and Aspect Based Sentiment Analysis

Team members

- Sravani Katlaganti
- Raja Tejasvi Prasad Panduga
- Yasmeen Haleem

Overview

- Introduction
- Problem Statement
- Methodology
- Dataset
- Exploratory Data Analysis
  - Data Preprocessing
  - Data Visualization
- Implementation
- Results
- Project Management
- References

Introduction

- **Innovative Sentiment Analysis Approach**: The approach leverages cutting-edge NLP techniques to perform aspect-based sentiment analysis, offering a more nuanced understanding of customer opinions in product reviews.
- **Focus on E-commerce Reviews**: Tailored specifically for e-commerce platforms, the project analyzes consumer reviews to extract sentiments associated with specific product features, such as battery life, camera quality, or price.
- **Utilization of Advanced Models**: The project employs state-of-the-art deep learning models like ALBERT and BERT, known for their effectiveness in processing and understanding natural language.

Introduction

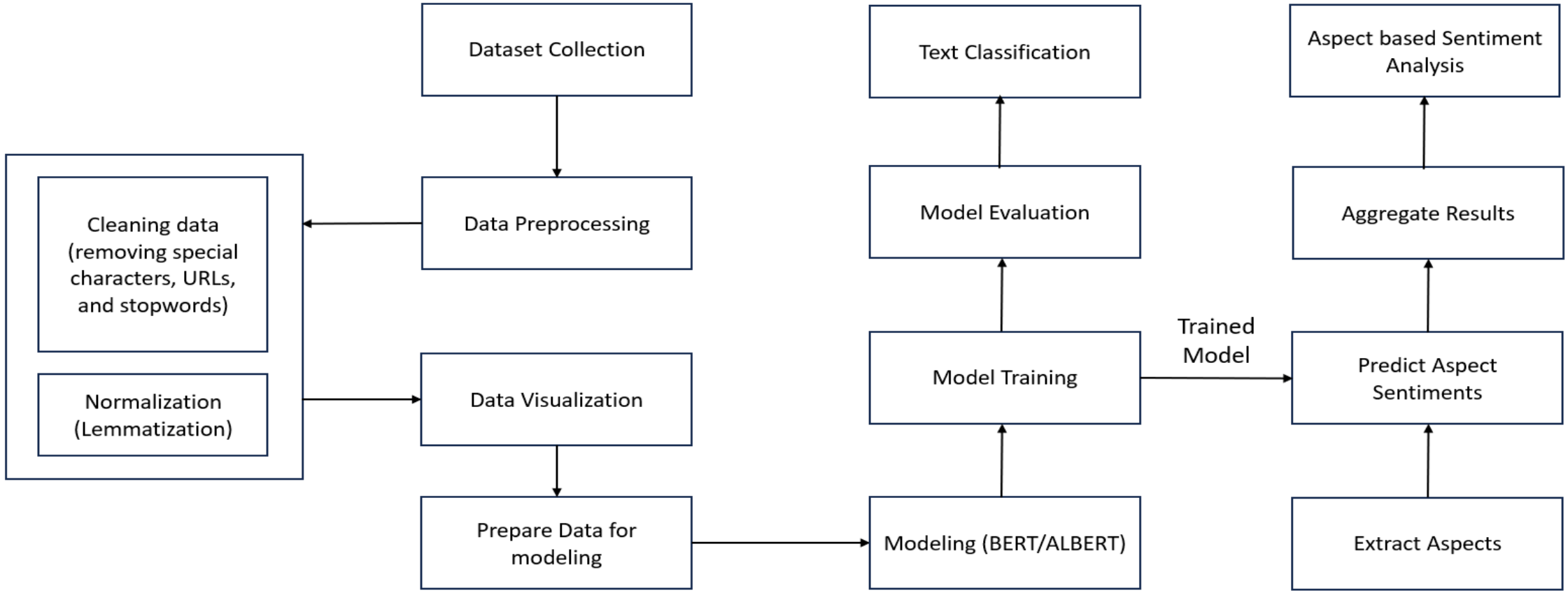
- **Aspect Extraction and Sentiment Classification**: It goes beyond traditional sentiment analysis by not only classifying overall sentiment but also identifying and evaluating sentiments related to individual aspects within the text.
- **Data-Driven Insights for Businesses**: By dissecting customer feedback into aspect-specific sentiments, this project provides valuable insights to businesses, helping them identify strengths and areas for improvement in their products.
- **Enhanced Customer Understanding**: The detailed analysis of the project aids in better understanding customer needs and preferences, leading to more informed product development and marketing strategies.
- **Contribution to Machine Learning and NLP Fields**: The project contributes to the fields of machine learning and NLP by addressing the complex task of aspect-based sentiment analysis, showcasing the application of advanced algorithms in real-world scenarios.

Problem Statement

- **Addressing the Need for Deep Analysis in E-commerce Reviews**: Traditional sentiment analysis tools are inadequate for the detailed, nuanced understanding required to extract aspect-specific sentiments from e-commerce reviews, which is crucial for truly understanding customer feedback.
- **Integrating Advanced Deep Learning Models for NLP**: The challenge lies in effectively harnessing state-of-the-art NLP models like ALBERT and BERT to analyze complex, varied customer reviews, a task that demands sophisticated natural language processing capabilities.
- **Transforming Raw Data into Actionable Business Insights**: There is a significant gap in translating the vast quantities of unstructured review data into specific, actionable insights for product improvement and targeted customer relationship strategies.
- **Enhancing Customer Experience through Targeted Analysis**: A key problem is the lack of deep customer understanding that businesses need for informed product development and marketing, which can be addressed by a system that accurately dissects customer sentiment on individual product aspects.

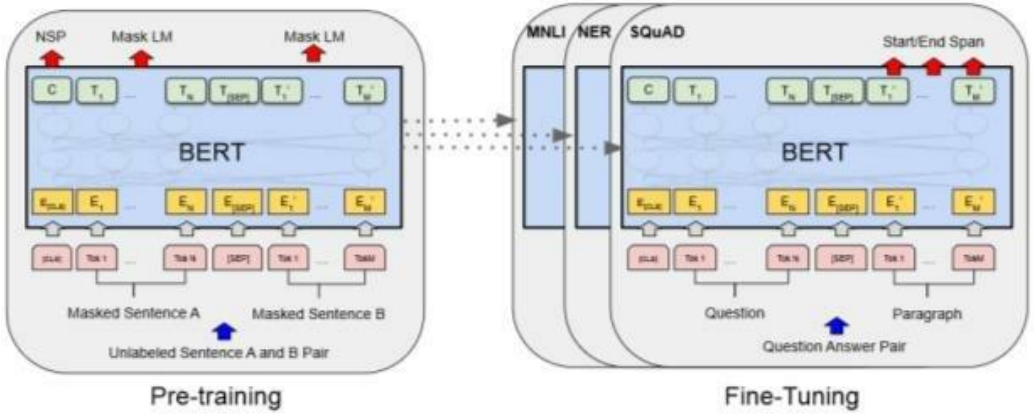
Methodology

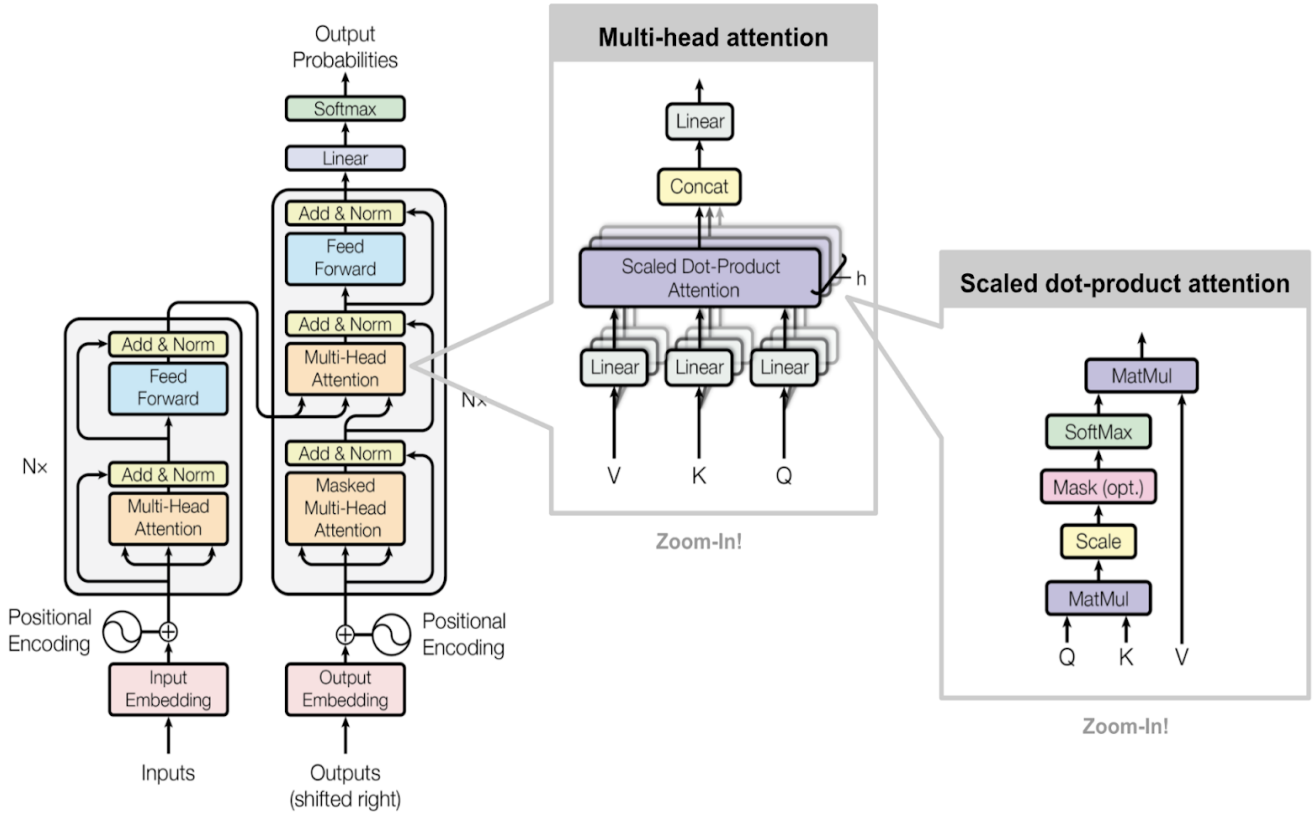
Workflow Diagram



Methodology

Bert Architecture



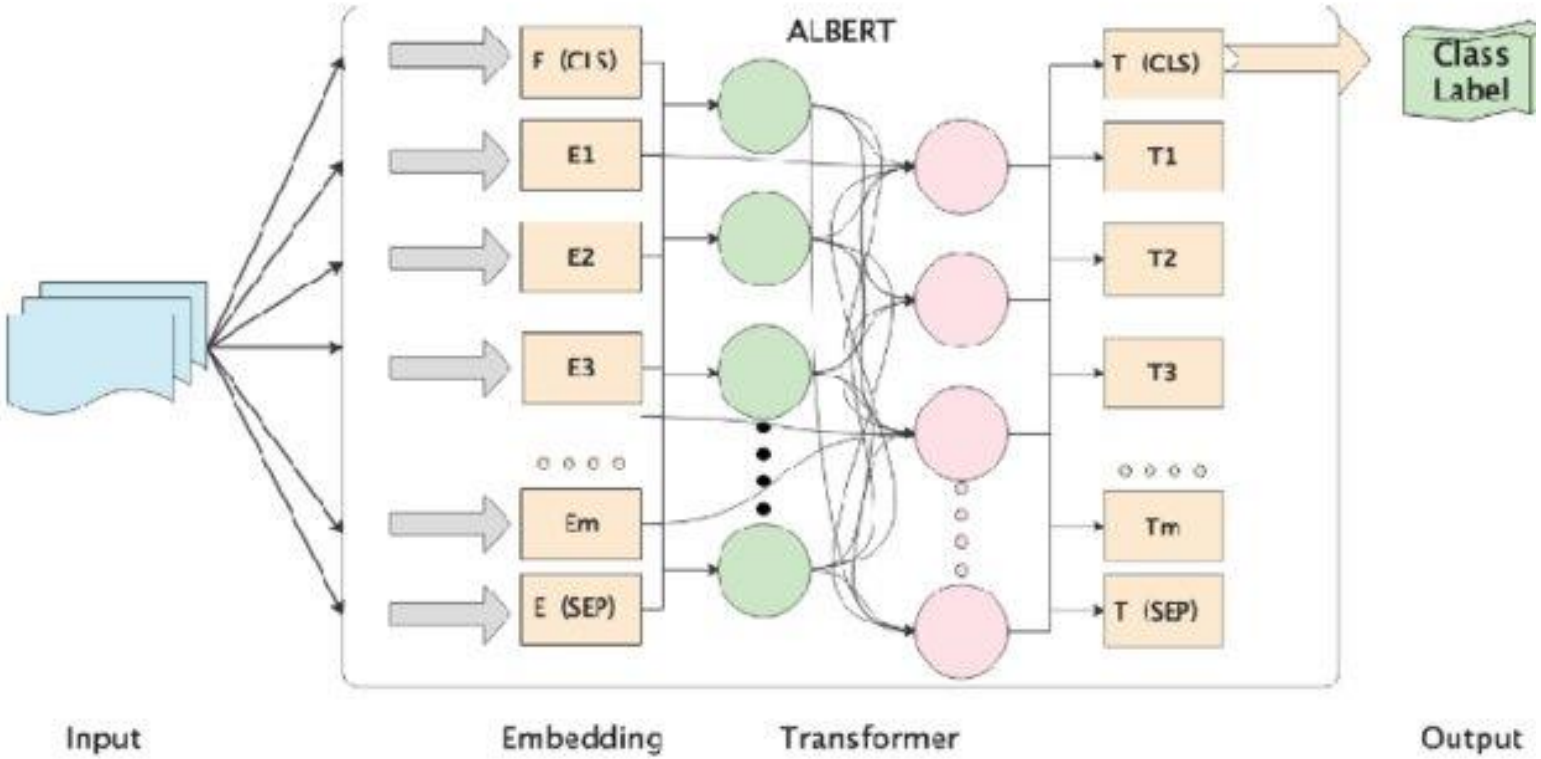


Methodology

- **Input Embedding and Positional Encoding**: The input text is converted into vectors (embeddings), which are then combined with positional encodings to retain the order of the words. This information is fed into the model to preserve the meaning based on the sequence of words.
- **Multi-Head Attention Mechanism**: This component of the architecture allows the model to focus on different parts of the input sequence for each prediction it makes. It does so by creating multiple attention mechanisms ('heads') that process the input in parallel, allowing the model to capture a richer understanding of context.
- **Scaled Dot-Product Attention**: Within the multi-head attention, this function calculates attention scores by scaling the dot product of the query and key vectors. It ensures that the softmax function has a stable gradient, as large values are scaled down before softmax is applied.
- **Feed-Forward Networks**: After attention has been applied, each position flows through a feed-forward neural network, which is applied identically to all positions. It consists of two linear transformations with a ReLU activation in between.
- **Output Probabilities**: The final step in the Transformer model includes a linear layer and a softmax function to produce probabilities for each word in the vocabulary. This output can be used for various tasks, such as language modeling, translation, or, in the case of BERT, masked language modeling and next sentence prediction.

Methodology

Albert Architecture



Methodology

- **Input Processing**: The input text is fed into the model where each token is embedded into a high-dimensional space. Special tokens like **[CLS]** for classification and **[SEP]** for separation are added to the sequence.
- **Embedding Layer**: The embeddings for each token, including the special **[CLS]** and **[SEP]**, are created to capture the semantic meaning of each word within the context of the sentence.
- **Transformer Blocks**: The core of ALBERT consists of repeating blocks of transformer layers. These layers use self-attention mechanisms to weigh the influence of different parts of the input text on each other and capture the context around each word.
- **Output Determination**: The output from the transformer layers, especially the transformed **[CLS]** token, is then used to predict the class label of the input sequence. This label could represent sentiment, categorization, or any other classification task.

Dataset

Amazon Product Reviews

- The Amazon reviews polarity dataset consists of reviews from amazon. We took this dataset from Kaggle.
- The data span a period of 18 years, including ~35 million reviews up to March 2013. It contains 34,686,770 Amazon reviews from 6,643,669 users on 2,441,053 products, from the Stanford Network Analysis Project (SNAP).
- This subset contains 1,800,000 training samples and 200,000 testing samples in each polarity sentiment.
- This dataset has feature columns of title, review text and label column of polarity.
- In the dataset, class 1 is the negative and class 2 is the positive. Each class has 1,800,000 training samples and 200,000 testing samples.

Dataset

Design of Features/Labels with diagram

	polarity	title	text
0	2	Stuning even for the non-gamer	This sound track was beautiful! It paints the ...
1	2	The best soundtrack ever to anything.	I'm reading a lot of reviews saying that this ...
2	2	Amazing!	This soundtrack is my favorite music of all ti...
3	2	Excellent Soundtrack	I truly like this soundtrack and I enjoy video...
4	2	Remember, Pull Your Jaw Off The Floor After He...	If you've played the game, you know how divine...

Exploratory Data Analysis

Data Preprocessing

Data Reading and Structure Preparation:

- CSV files containing Amazon reviews are read into pandas DataFrames.
- Data structure is known using info() and describe() functions.
- The shape of the train and test data is checked and displayed.
- The DataFrames are truncated to handle the first 50,000 and 5,000 rows for the train and test sets, respectively.
- Columns are renamed to 'polarity', 'title', and 'text' for clarity.

Concatenation of Columns:

- The 'text' and 'title' columns are concatenated to form a single text column.
- After concatenation, the 'title' column is dropped from the DataFrame.

Data Preprocessing

Cleaning:

- All text is converted to **lowercase** to maintain uniformity and to ensure that the algorithm treats words with the same root similarly regardless of their case.
- Text in square brackets is removed, which might include references or hyperlinks.
- Links are removed, which are not useful for sentiment analysis.
- **Punctuation** is removed, as it often does not contribute to the sentiment of the text.
- Words containing numbers are removed, as they are likely not useful for sentiment analysis.
- **Stop words** (commonly used words that do not carry significant meaning, like 'the', 'is', etc.) are removed to focus on words that carry the sentiment.
- **Extra whitespace** is removed to clean the text further.

Data Preprocessing

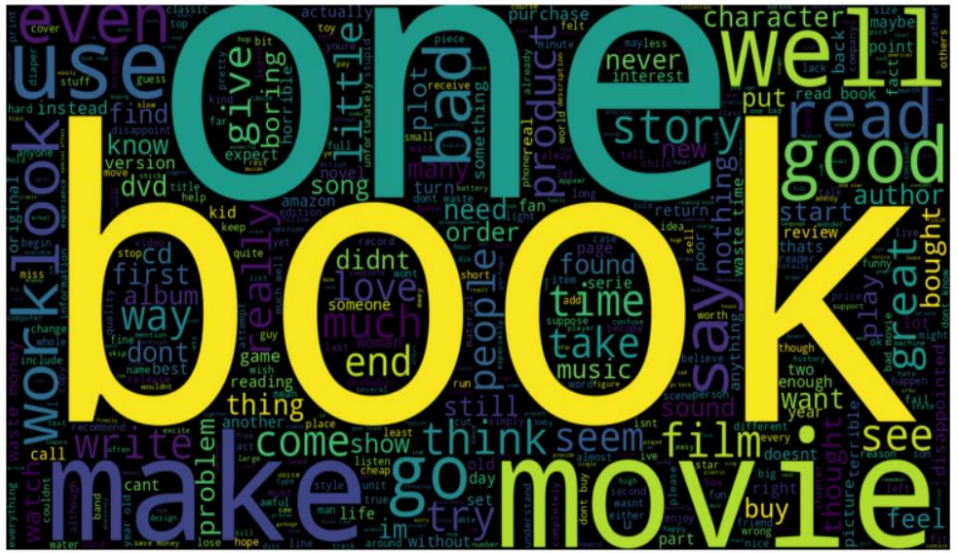
Normalization:

- Lemmatization is performed, which reduces words to their base or root form (lemma).
- **Parts of speech** for words are determined to improve the accuracy of the lemmatization process.
- **Tokenization** and **lemmatization** are applied to the 'text' column, converting words into their lemmatized form based on their identified part of speech.

Data Visualization

Word Clouds

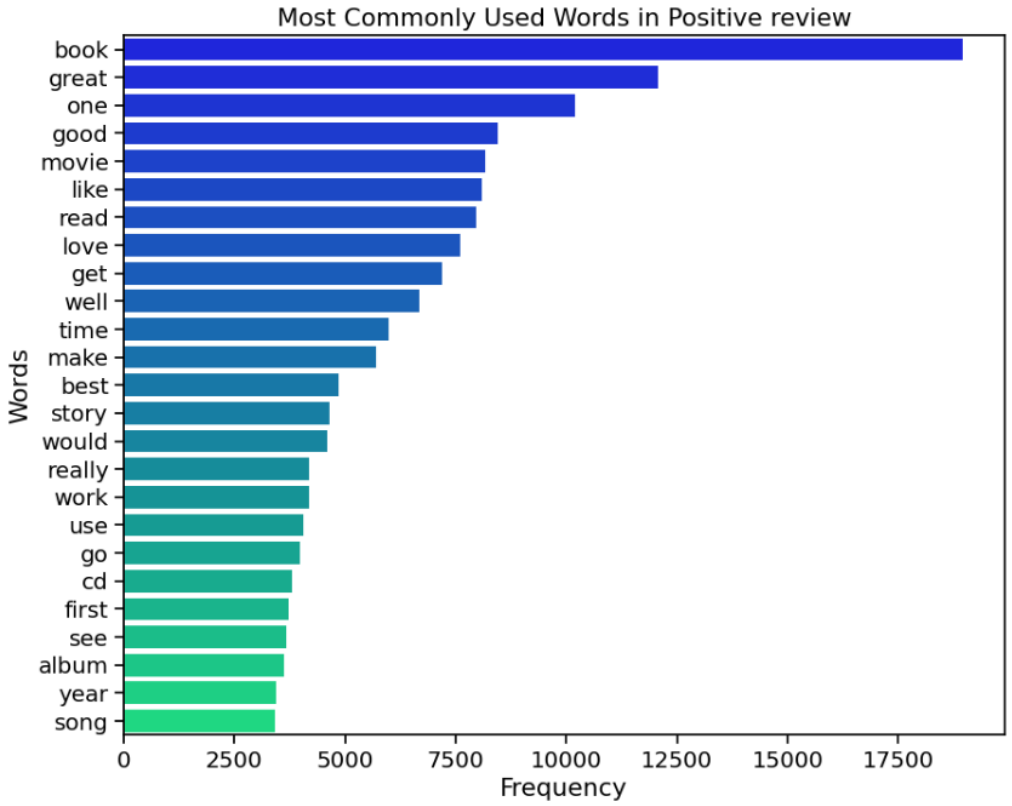
Positive Reviews Word Cloud

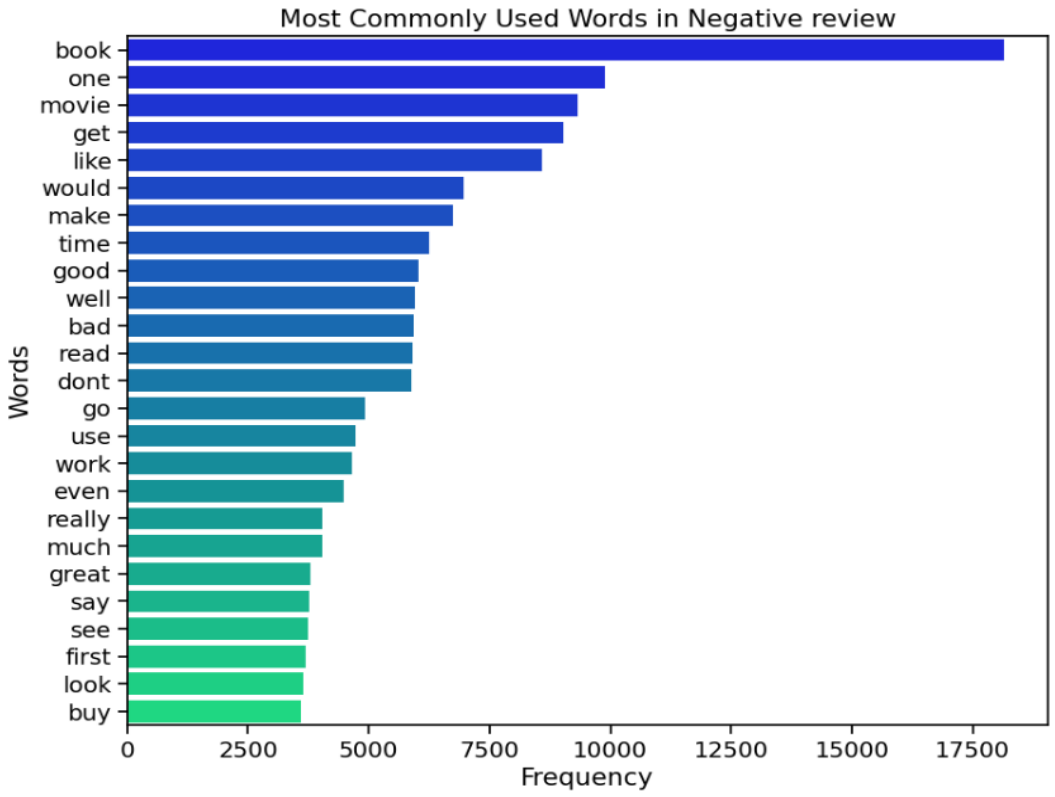


Negative Reviews Word Cloud

Data Visualization

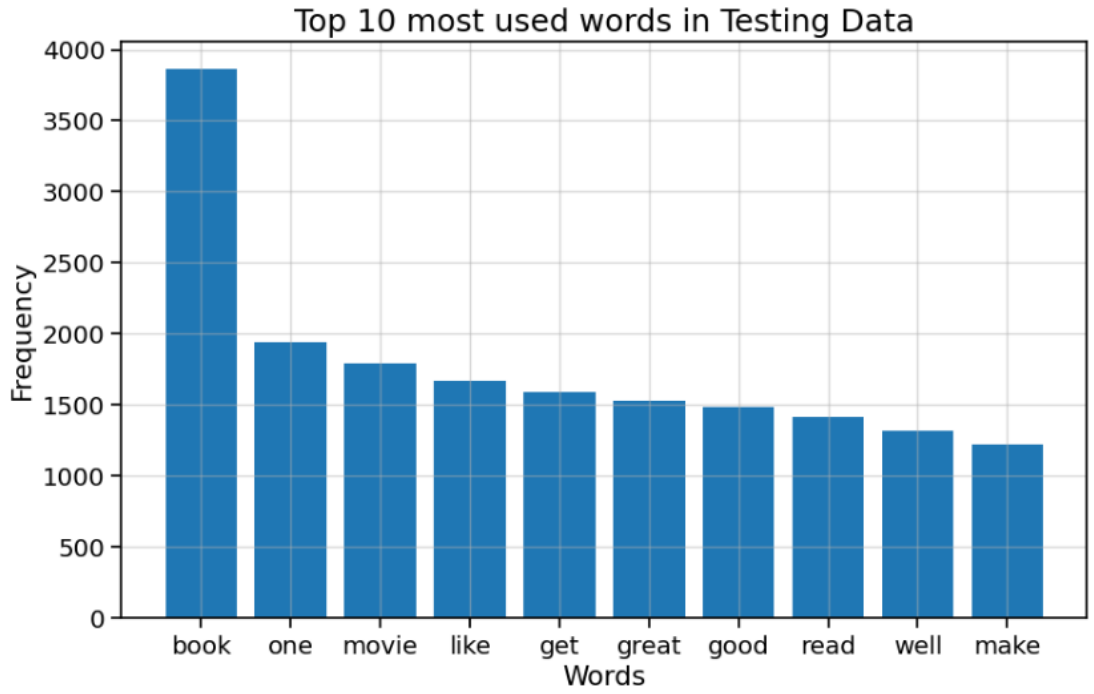
Frequency Distribution





Data Visualization

Top 10 most used words



Implementation

Algorithm: Aspect-Based Sentiment Analysis

Inputs: train_data: Training dataset with reviews, test_data: Testing dataset with reviews

Outputs: aspect_sentiments: Sentiment predictions for each aspect in the reviews

Steps:

1. Read and preprocess the input datasets:
    - Load the training and testing data from specified file paths.
    - Concatenate the title and text columns for both training and testing data.
    - Clean the concatenated text by removing special characters, URLs, and stopwords.
    - Normalize the cleaned text using lemmatization.
2. Visualize data:
    - Generate word clouds for positive and negative sentiments.
    - Plot the frequency distribution of words in the reviews.
    - Create bar plots for the most used words in the training and testing datasets.

Implementation

3. Prepare the data for model training:
    - Initialize the tokenizer from the BERT/ALBERT pre-trained model.
    - Tokenize the text of the training and testing data.
    - Convert the tokenized data into tensors suitable for model input.
4. Set up the data loaders:
    - Create TensorDatasets from the input and mask tensors, and labels.
    - Initialize DataLoaders for batching the data for training and evaluation.
5. Initialize the BERT/ALBERT model:
    - Load the BERT/ALBERT pre-trained model suitable for sequence classification.
    - Move the model to the GPU for faster computations.
6. Train the BERT/ALBERT model:
    - Define the optimizer with an appropriate learning rate.
    - Iterate over the training DataLoader and perform forward and backward passes.
    - Compute the loss and update the model parameters.

Implementation

7. Evaluate the model:
    - Set the model to evaluation mode.
    - Predict sentiments on the testing dataset.
    - Calculate accuracy, precision, recall, and F1 score from the predictions.
8. Perform Aspect-Based Sentiment Analysis:
    - Extract aspects (noun chunks) from the reviews using spaCy.
    - Predict the sentiment for each extracted aspect using the trained BERT/ALBERT model.
    - Store the sentiment predictions corresponding to the aspects.
9. Aggregate and display the results:
    - Iterate over the processed reviews and aspects.
    - Print the review text, aspects, and corresponding sentiment predictions.

Implementation

Implementation libraries

- **Transformers:** A library by Hugging Face that provides general-purpose architectures for Natural Language Understanding (NLU) and Natural Language Generation (NLG).
- Usage: Accessing pre-trained BERT and ALBERT models and their respective tokenizers for natural language processing tasks.
- **Sklearn:** A machine learning library for Python, used for splitting the dataset and evaluating the model.
- Usage: Splitting the dataset into training and testing sets, calculating performance metrics (accuracy, precision, recall, F1-score),and performing additional machine learning tasks if needed.
- **Pandas:** A library providing high-performance, easy-to-use data structures, and data analysis tools.
- Usage: Loading and preprocessing the dataset before it is fed into the models
- **NumPy:** A library for the Python programming language, adding support for large, multi-dimensional arrays and matrices.
- Usage: Performing operations on numerical data, supporting Pandas operations, and sometimes used for handling the outputs of the models.

Implementation

Explanation of Implementation

- **Data Preprocessing:** The algorithm starts by loading the datasets and then concatenates the title and body of the reviews to have a complete context. It cleans the text by converting it to lowercase, removing special characters, links, numbers, and stopwords. This is essential for reducing noise in the data and ensuring that the machine learning model focuses on the meaningful content of the reviews.
- **Data Visualization:** Visualizing the data helps understand the distribution of words and sentiments within the reviews. Word clouds show the most frequent words in positive and negative reviews, while frequency plots reveal the most common terms. These visual aids can help identify prevalent themes or issues discussed in the reviews.
- **Preparation for Model Training:** Tokenization involves converting the text into a format that the model can understand, which in this case is a series of tokens or word IDs. The BERT/ALBERT tokenizer is used here for its efficiency in understanding the context of words in sentences.
- **Data Loaders Setup:** Data loaders are configured to automate the batching process during training and evaluation. They ensure that data is fed to the model in manageable sizes and in an orderly fashion, which is crucial for efficient training.

Implementation

Explanation of Implementation

- **ALBERT Model Initialization:** The BERT/ALBERT (A Lite BERT) model is initialized for sequence classification. ALBERT is a variant of BERT that is optimized for lower memory consumption and faster training. The model is transferred to a GPU to leverage accelerated hardware for training.
- **Model Training:** The optimizer is set up with a learning rate to update the model's weights. During training, the model's parameters are adjusted to minimize the prediction error, using the provided sentiment labels as the ground truth.
- **Model Evaluation:** In evaluation mode, the model's performance is tested against unseen data. This phase involves calculating various metrics like accuracy, precision, recall, and F1 score to quantify the model's performance.
- **Aspect-Based Sentiment Analysis:** Aspect extraction is conducted using spaCy to identify the different aspects (or features) mentioned in each review. The sentiment of each aspect is then predicted using the trained BERT/ALBERT model.
- **Results Aggregation and Display:** Finally, the algorithm iterates over the dataset, prints out each review, and lists the aspects along with their predicted sentiments. This step is key to understanding the detailed sentiment toward various aspects of the products being reviewed.

Results: Performation Metrics

BERT Model

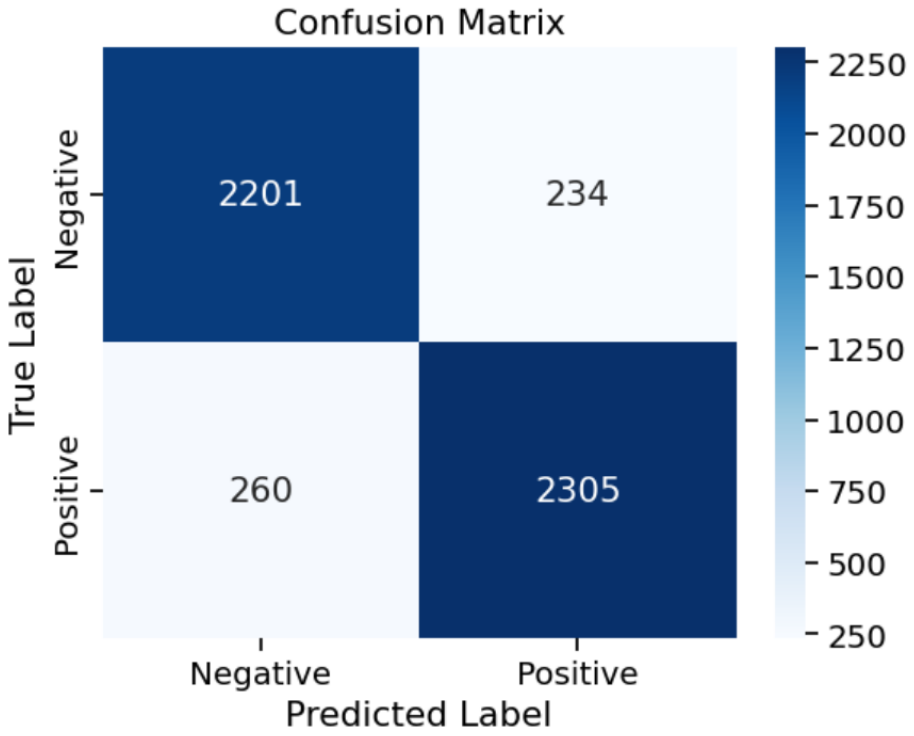
Accuracy: 0.9012
Precision: 0.9078377313903111
Recall: 0.898635477582846
F1 Score: 0.9032131661442007

ALBERT Model

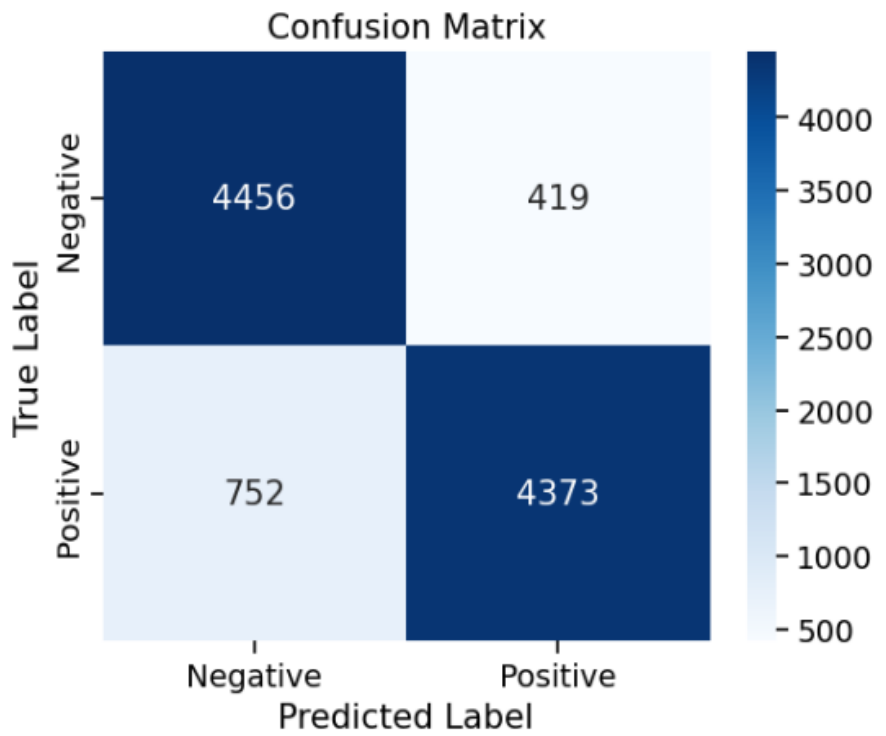
Accuracy: 0.8829
Precision: 0.9125626043405676
Recall: 0.8532682926829268
F1 Score: 0.8819199354643541

Results: Visualization

BERT Model

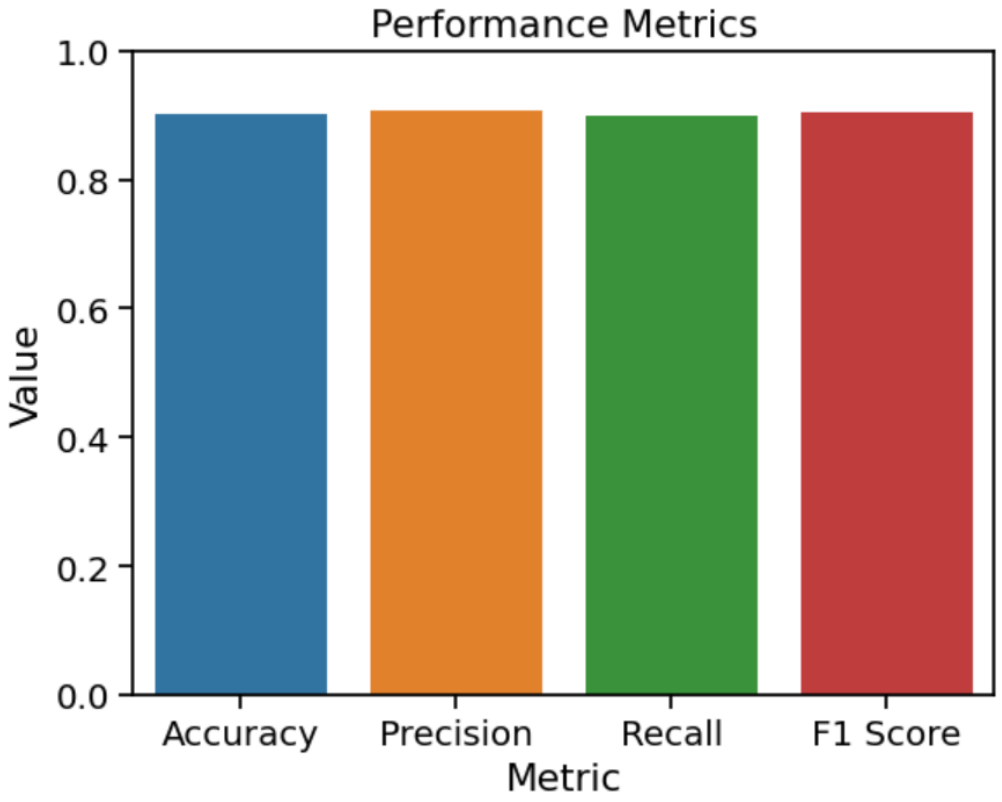


ALBERT Model

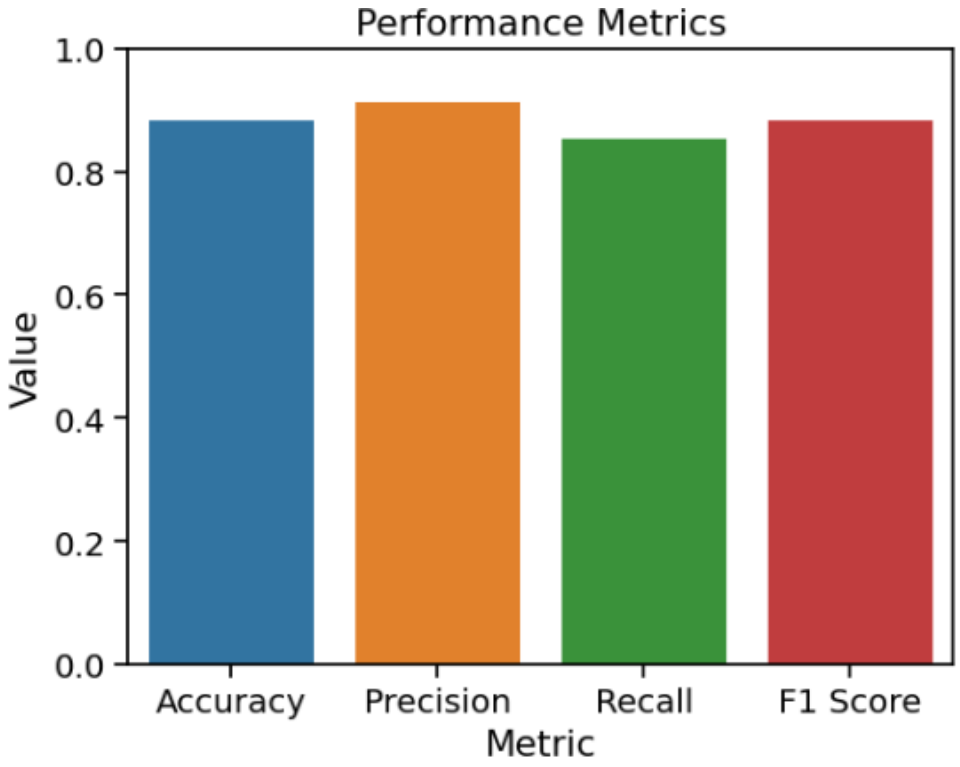


Results: Visualization

BERT Model

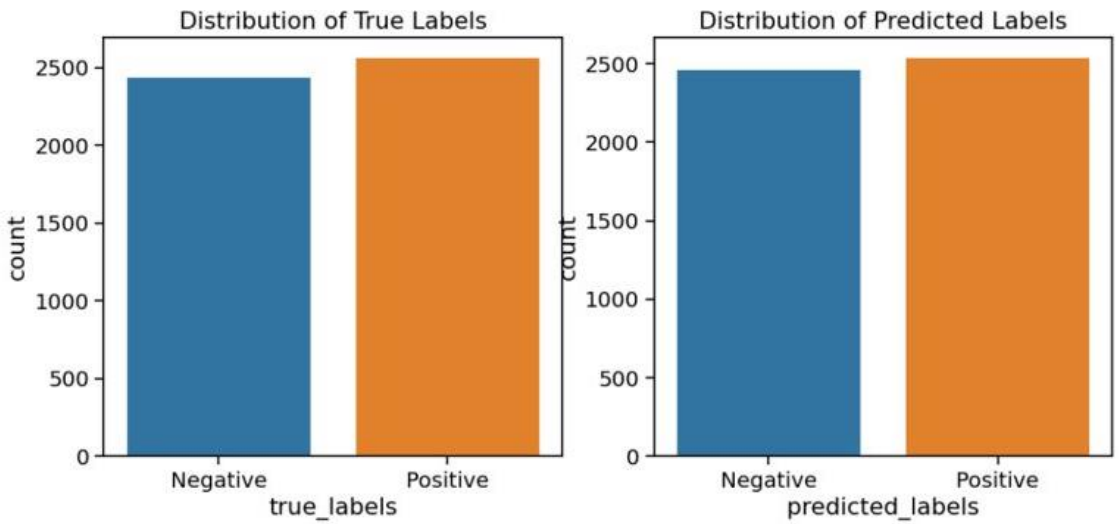


ALBERT Model

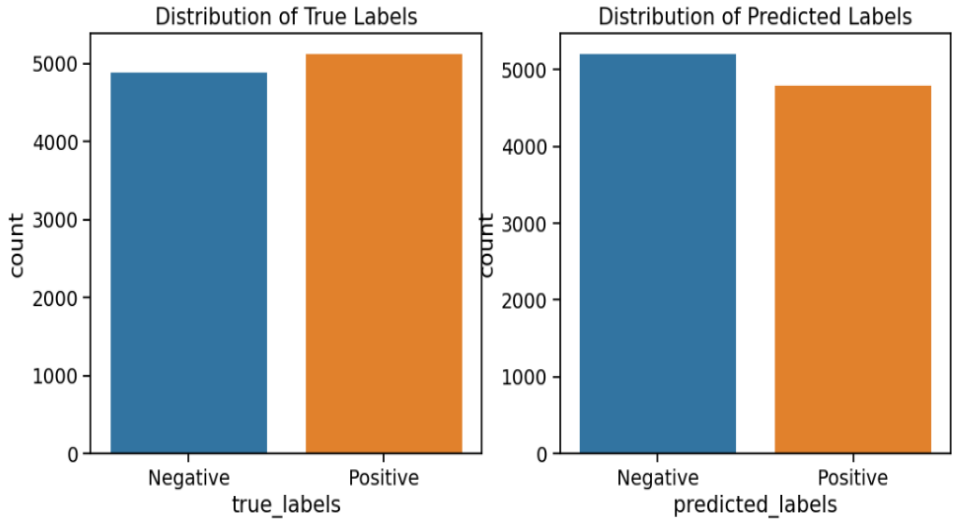


Results: Visualization

BERT Model



ALBERT Model



Project Management

Module	Description	Action item	Member	Percentage
1	BERT Model	1. Concatenating 2. Text cleaning 3. Normalization(Lemmatization) 4. Data Visualization(Word clouds, Frequency distribution, Bar plots) 5. Bert Model implementation 6. PPT slides	Sravani Katlaganti	34%
2	ALBERT Model	1. Concatenating 2. Text cleaning 3. Normalization(Lemmatization) 4. Data Visualization(Word clouds, Frequency distribution, Bar plots) 5. Albert Model implementation 6. PPT slides	Raja Tejasvi Prasad Panduga	33%
3	Aspect based Sentimental Analysis	1. Aspect Extraction 2. Implementing trained BERT/ALBERT model for aspect based sentimental analysis 3. Results Aggregation 4. Model Comparison and analysis 5. Output Analysis 6. PPT slides	Yasmeen Haleem	33%

Issues

- Dataset contains around 35 million reviews which is huge and complex to handle and run.
- Training and fine-tuning BERT and ALBERT models required substantial computational resources like high-end GPUs or TPUs, which didn't efficiently work in Google Colab.
- Hence, we opted to run in Kaggle platform which is better, but it is also time taking.

Literature Survey

ALBERT: A Lite BERT for Language Understanding

ALBERT (A Lite BERT) is a state-of-the-art natural language processing (NLP) model developed by Google AI in 2020. Building upon the success of BERT, ALBERT offers several advantages, including:

Reduced Parameter Size and Computational Cost: ALBERT achieves similar performance to BERT while having significantly fewer parameters, making it more efficient for training and deployment on resource-constrained devices (Sun et al., 2020).

Improved Sentence Ordering Prediction: ALBERT incorporates a sentence ordering task into its pre-training, which enhances its ability to capture inter-sentence relationships and improve performance on tasks like question answering and natural language inference (Rajakumar et al., 2020).

Effective for Sentiment Classification: ALBERT has demonstrated strong performance in sentiment classification tasks, achieving state-of-the-art results on benchmark datasets like GLUE and SST-2 (Sun et al., 2020). This is attributed to its ability to capture subtle nuances in language and extract sentiment information effectively.

Adaptability through Fine-Tuning: Similar to BERT, ALBERT can be fine-tuned for specific sentiment analysis tasks, allowing it to tailor its parameters to the domain and characteristics of the dataset (Sun et al., 2020).

Overall, ALBERT presents a valuable addition to the NLP landscape, offering a balance between performance, efficiency, and adaptability. Its effectiveness in sentiment classification makes it a compelling choice for various applications, particularly in resource-constrained environments.

References

1. https://thesai.org/Paper_3Sentiment_Analysis_on_Amazon_Product_Reviews.pdf

2. https://towardsdatascience.com/sentiment-analysis-on-amazon-reviews-45cd169447ac

3. https://www.kaggle.com/datasets/kritanjalijain/amazon-reviews

4. https://ieeexplore.ieee.org/document/9402414

5. https://github.com/joshivaibhav/AmazonCustomerReview/blob/master/amazondata.csv

6. Sun, Y., Sagun, L., Young, T., & Dubey, S. (2020). ALBERT: A Lite BERT for Language Understanding. arXiv preprint arXiv:2004.02324. 7.

7. Rajakumar, G., Szekely, P., D'Souza, S., Pang, L., & Neubig, G. (2020). Do Transformers Really Need Positional Encodings? An Analysis of BERT and ALBERT. arXiv preprint arXiv:2003.15581.

Thank you

Links

- https://drive.google.com/drive/folders/1MuMFqWfMhRbrjN-Eec27_t6MlcR1CNT_?usp=sharing