6CS012-Artificial Intelligence and Machine Learning

Academic Year	Module	Assessment	Type
2024	Artificial Intelligence and Machine Learning (6CS012)	Text Classification	Individuals

Student Id: NP03CS4S220230

University ID: 2227436

Group: L6CG2

Student Name: Abhinav Niraula

Module Leader: Mr. Siman Giri

Instructor: Miss. Sunita Parajuli

Abstract

This study's sentiment analysis model performs admirably in reliably extracting sentiments from textual material. After a thorough assessment using test data that hasn't been seen yet, the model obtains an impressive test accuracy of about 80.56%, highlighting its strong categorization abilities. The model's ability to capture subtle patterns of sentiment is further validated by balanced precision, recall, and F1-score for both positive and negative feelings. The model has a slightly elevated false positive rate; nonetheless, it consistently strikes a balance between accurately detecting positive attitudes and reducing misclassifications. These results highlight the model's dependability and efficiency in practical sentiment analysis applications, offering insightful information for decision-making procedures in a variety of fields. The model's performance may be improved through more optimization efforts, opening the door for a wider deployment and utility in

Table of Contents

Contents

Table of Figures

1. Introduction

The dataset consists of Amazon Book Reviews, which encompass either summaries accompanied by ratings or user-generated reviews. In accordance with the substance of the text, text classification involves the organization of data into classes. Based on the sentiments conveyed in the review text, the objective of this endeavor is to classify Amazon book reviews into groups, such as positive or negative.

1.1. Aims and Objectives

1.1.1. Aims

- By integrating explainable AI techniques, augment the interpretability of sentiment analysis outcomes in Amazon book reviews.
- To enhancing the robustness of the model, examine how the length of reviews and the manner of writing impact the accuracy of sentiment classification.
- To augment the predictive capabilities of the sentiment analysis model, investigate ensemble learning methodologies, including layering and boosting.

1.1.2. Objectives:

- To offer insights into the manner in which particular words or phrases impact predictions in Amazon book reviews, it is imperative to incorporate explainable AI methods.
- To maximise model performance and detect patterns, it is necessary to perform an analysis of the correlation between writing style features (such as readability scores and sentiment intensity), review duration, and sentiment classification accuracy.
- To leverage the collective strength of multiple base models in capturing

diversified aspects of sentiment conveyed in Amazon book reviews, ensemble learning techniques should be implemented to combine predictions.

1.2. Sequential Model Architecture

The sequential model setup for text classification begins with defining the text data, ensuring that all parts are the same length. The words are then transformed into mathematical figures using an embedding layer. The text is then understood using a special sort of recurrent neural network known as bidirectional LSTM, which takes into account both past and prospective context. To compress the input, a global max pooling layer is inserted, followed by additional layers that detect complicated patterns and use L2 regularization techniques to avoid overfitting. The final layer estimates the likelihood of favourable sentiment. Once the model has been configured with appropriate loss and optimization parameters, it is trained and periodically saved using callbacks for future usage. This system efficiently manages text data for sentiment analysis.

1.3. Outline of Report

The report begins with an introduction that sets the stage by describing the project's background and the dataset used. It then digs into the approach, explaining how the information was pre-processed, the computational architecture created with TensorFlow and Keras, and the strategies used to assess the model's performance. The results will be presented next, highlighting major discoveries such as model precision, graphs of train vs validation loss, confusion matrices, and the ROC curve. Following that, a discussion and analysis section examines the results' strengths, shortcomings, and prospective areas for improvement. Finally, the paper finishes with a summary, which summarizes the project's accomplishments and suggests areas for further research.

2. Methodology

Data cleaning involved initial preprocessing steps on the Amazon Book Review dataset. This included punctuation removal, text lowercase conversion, stopwords elimination, and lemmatization using NLTK. These actions standardized the text for subsequent modelling.

The model architecture employed a sequential neural network with TensorFlow and Keras. It featured an embedding layer for dense numerical representation conversion, followed by a bidirectional LSTM layer capturing contextual information bidirectionally. Additionally, a global max pooling layer reduced dimensionality, and multiple dense layers with ReLU activation and dropout regularization managed complex patterns, preventing overfitting. The final layer used sigmoid activation to predict positive sentiment probability.

Training utilized the Adam optimizer and binary cross-entropy loss function, with various hyperparameters adjusted for optimal performance. These included LSTM units, dropout rates, batch size, epochs, and L2 regularization for dense layers, minimizing overfitting.

Overall, the methodology involved data preprocessing, model architecture design, training, and hyperparameter optimization, culminating in an effective sentiment analysis model for Amazon book reviews.

2.1. Data Pre-Processing

Several strategies were used in the very first information preparation phase to improve and arrange the verbal data for further analysis. To protect text integrity, punctuation marks were deleted first using particular patterns. After that, all text was transformed into lowercase letters to ensure consistency and avoid conflicts caused by case variances. Lemmatization was used with the NLTK package to reduce phrases to their base forms, improving consistency while decreasing vocabulary complexity. This method helps to handle different word forms as a single entity, which improves the accuracy of future studies. A word cloud visualization (Vu, 2023) was also created to represent word distribution, providing insights into the degree of appearance from most to least frequent. To simplify further modelling operations, TensorFlow Keras' Tokenizer class was used to break down text onto individual tokens and transform them to numerical sequences. Furthermore, padding was used to ensure that the sequence length was consistent, which is critical for good neural network training. These rigorous data cleaning and preparation techniques guarantee that the written data is improved, standardized, and ready for subsequent analytical and modelling operations.

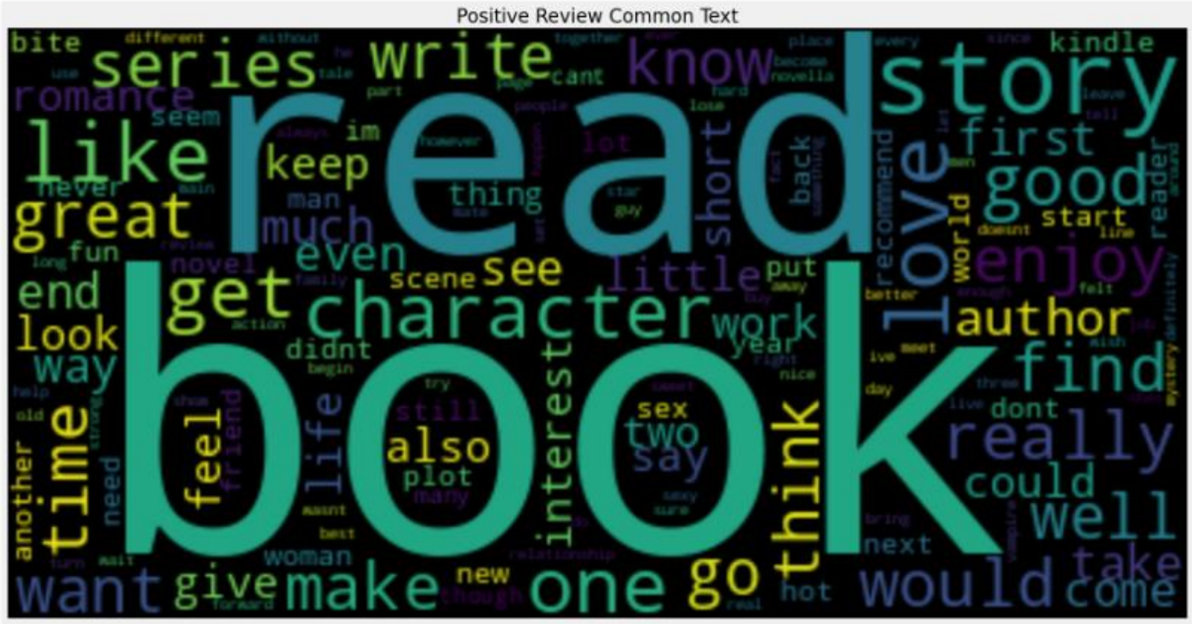


Figure 1: Word Cloud

2.2. Model Building

2.2.1. Model Summary:

This model summary presents a deep learning architecture intended for tasks related to sentiment analysis. This is an explanation:

2.2.1.1. Embedding Layer

- Converts the input vocabulary into an embedding space, which is a lower-dimensional space.
- The vocabulary's size is the input dimension.
- Dimension of output: 128.
- 120 is the input length.

2.2.1.2. Bidirectional LSTM Model:

- Applies bidirectional long short-term memory (LSTM) cells to the capture of forward and backward temporal interdependence.
- 64 units are used.

- Brings back sequences Indeed, in order to move sequences to the following layer.
- 20% dropout rate (to avoid overfitting).

2.2.1.3. Global Max Pooling Layer

- Extracts the most notable characteristics from various temporal sequences.

2.2.1.4. Dense Layers

- Refinement of features and non-linear transformations are carried out.
- 32 units are used.
- Rectified Linear Unit (ReLU) is used to activate.
- Regularization: 0.07 parameter L2 regularization.

2.2.1.5. Dropout Layer

- 5% of neurons are randomly deactivated during training to help prevent overfitting.

2.2.1.6. Output Layer

- Dense layer that supports binary classification (such as sentiment polarity prediction) and has a single neuron and sigmoid activation.

2.2.1.7. Model Compilation:

- Loss Function: Binary Cross-Entropy
- Optimizer: Adam
- Metrics: Accuracy

2.2.1.8. Callbacks

- Model CheckPoint is used to save the best accuracy.

```
Model: "sequential_7"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_7 (Embedding)     (None, 120, 128)          3988992

 bidirectional_7 (Bidirecti  (None, 120, 128)          98816
 onal)

 global_max_pooling1d_7 (Gl  (None, 128)               0
 obalMaxPooling1D)

 dense_42 (Dense)            (None, 32)                4128

 dense_43 (Dense)            (None, 32)                1056

 dense_44 (Dense)            (None, 32)                1056

 dense_45 (Dense)            (None, 32)                1056

 dense_46 (Dense)            (None, 32)                1056

 dropout_7 (Dropout)         (None, 32)                0

 dense_47 (Dense)            (None, 1)                 33

=================================================================
Total params: 4096193 (15.63 MB)
Trainable params: 4096193 (15.63 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 2: Model Summary

2.2.2. Training The Model

The binary cross-entropy loss function, which is frequently employed for binary classification applications like sentiment analysis, was utilized to train the model. Because of its effective optimization skills and flexible learning rate, the Adam optimizer was used for training. The "epochs=4" option in the training code block indicates that the model was trained for 4 epochs in each iteration. The train-validation loss curve illustrates the training behavior, which is characterized by an initial drop in both training and validation loss, suggesting that the model is learning. But after a few epochs, the training loss keeps going down but the validation loss appears to be slightly increasing. The model may be memorizing the training if there is a gap between the validation loss and the training. The following plots between the accuracies and between the losses shall display the real sample of the whole thing. (Awan, 2022)

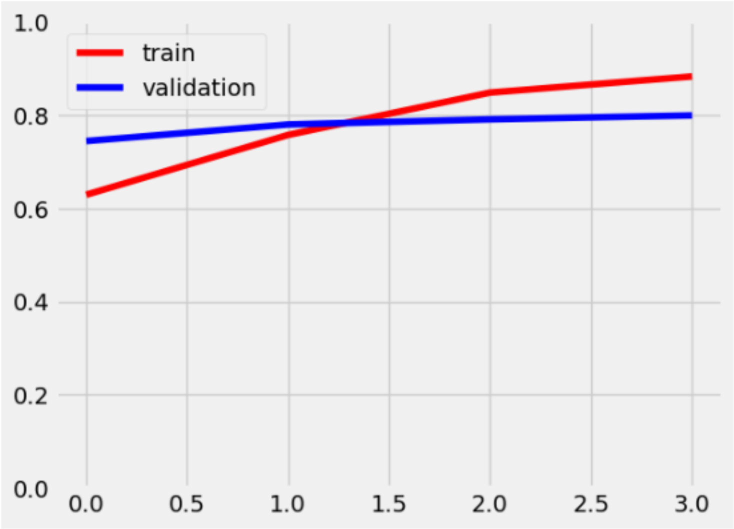


Figure 3: Validation Graphs

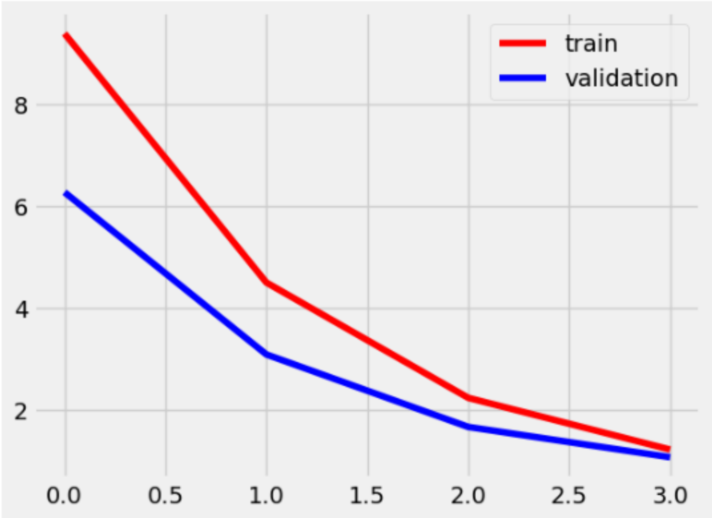


Figure 4: Loss Graphs

2.2.3. Evaluating the Model

Promising performance is shown when the sentiment analysis model is evaluated on test data that has not yet been seen. The model shows that it can accurately classify sentiments, with a test accuracy of about 80.56%. This is further supported by the classification report, which displays balanced F1-score, precision, and recall (Agarwal, 2024) for both positive and negative classes, with values ranging from 0.78 to 0.83. This suggests that a sizable percentage of positive occurrences are captured, and that true positive and false positive forecasts are well balanced. The confusion matrix shows how well the model performed in accurately identifying both positive and negative attitudes by breaking down predictions against real labels in great detail. The model performs well overall in sentiment categorization on the unseen test, however there are somewhat more false positives than false negatives.

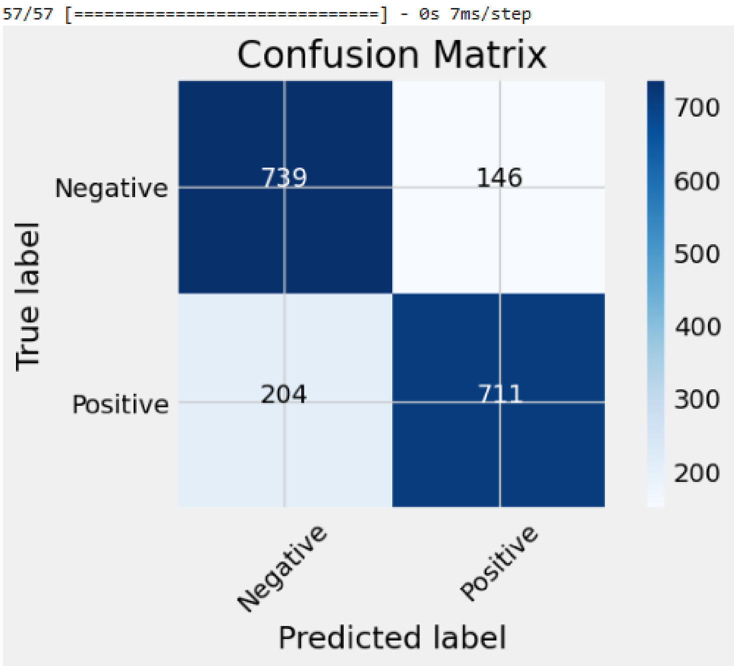
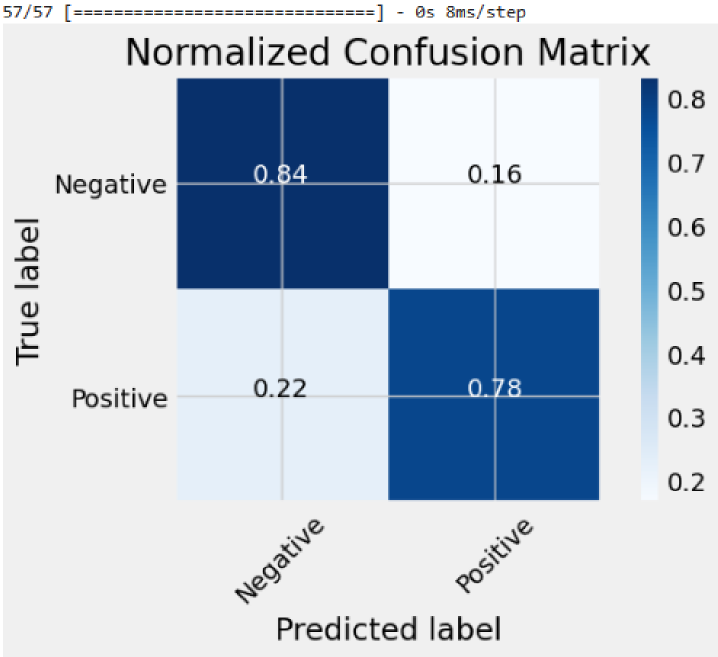


Figure 5: Confusion Matrix

```
57/57 [==============================] - 0s 8ms/step
```



	precision	recall	f1-score	support
0	0.78	0.84	0.81	885
1	0.83	0.78	0.80	915
accuracy			0.81	1800
macro avg	0.81	0.81	0.81	1800
weighted avg	0.81	0.81	0.81	1800

Figure 6: Evaluation Metrics

3. Final Discussions

We can draw the conclusion that the sentiment analysis model exhibits strong skills in accurately classifying sentiments based on the evaluation metrics and the model's performance on the unseen test data. The test accuracy of the model is roughly 80.56%, and it exhibits consistent performance across many assessment measures, with balanced precision, recall, and F1-score for both positive and negative classes. The model accurately detects positive attitudes while minimizing misclassifications, while having a somewhat greater proportion of false positives than false negatives. Overall, these findings demonstrate the model's efficacy and dependability in sentiment analysis tasks, suggesting that it could be used in practical applications where it is essential to recognize and evaluate the sentiments included in textual material. The following ROC-AUC curve is a testament to the above statements:

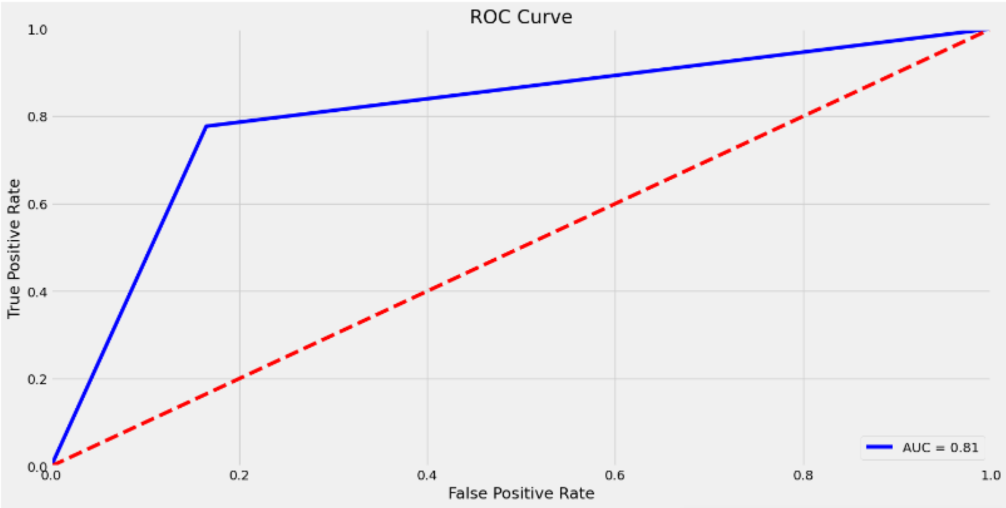


Figure 7: ROC-AUC Curve

References

Agarwal, S., 2024. *Evaluation Metrics For Classification Model - Analytics Vidhya.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/
[Accessed 12 May 2024].

Awan, A., 2022. *Recurrent Neural Network Tutorial (RNN)* |*DataCamp.* [Online]
Available at: https://www.datacamp.com/tutorial/tutorial-for-recurrent-neural-network?/=&irclickid=1E7QoMxxrxyKWW3WNjWFFSeGUkHQASQoD2R0180&irgwc=1&utm_medium=affiliate&utm_source=impact&utm_campaign=000000_1-1260174_2-mix_3-all_4-na_5-na_6-na_7-mp_8-affl-ip_9-na_10-bau
[Accessed 12 May 2024].

Vu, D., 2023. *Python WOrd Clouds Tutorial: How to Create a Word Cloud* | *DataCamp.* [Online]
Available at: https://www.datacamp.com/tutorial/wordcloud-python
[Accessed 12 May 2024].