Sentiment Analysis

# Objective

The objective of this project is to perform sentiment analysis on textual data using a Recurrent Neural Network (RNN). The goal is to determine whether a given sentence expresses a positive or negative sentiment with high accuracy.

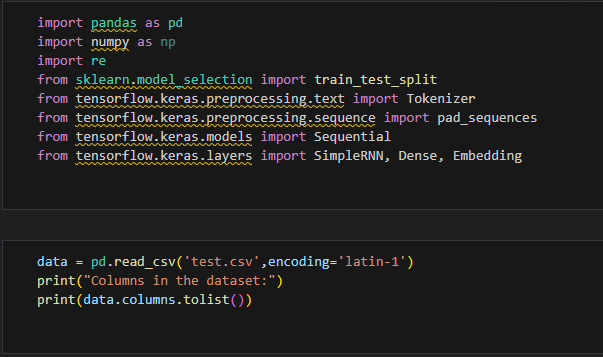
# Dataset Description

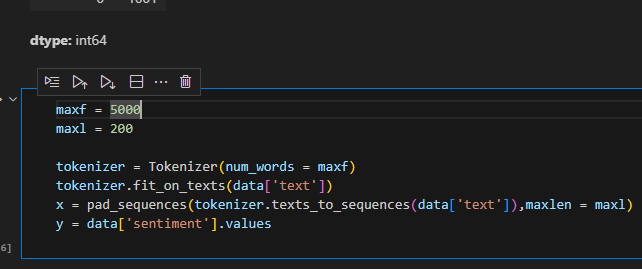
The dataset used includes movie reviews with corresponding sentiment labels (positive or negative). Each review is a sequence of words which are preprocessed and tokenized before being passed to the model.

# Model Overview

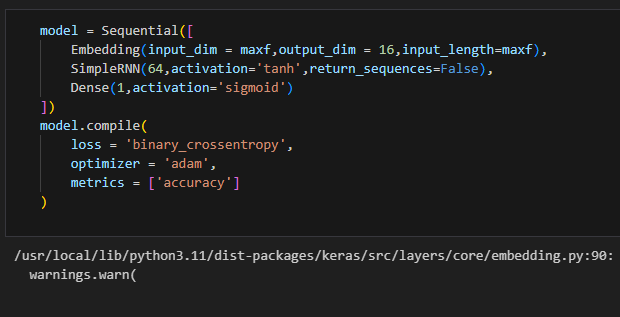
The model implemented is a simple RNN model built. It consists of an embedding layer, a single RNN layer, and a fully connected linear layer that outputs a probability score between 0 and 1, indicating the sentiment polarity.

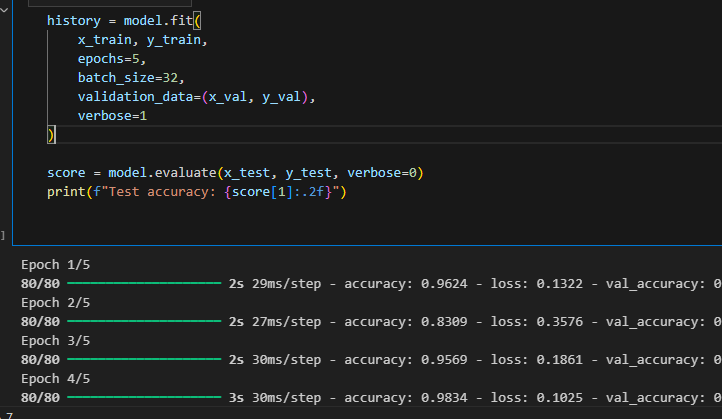
# Python Code Summary

  
1. Load and preprocess the text dataset.  
2. Tokenize and pad sequences to a fixed length.

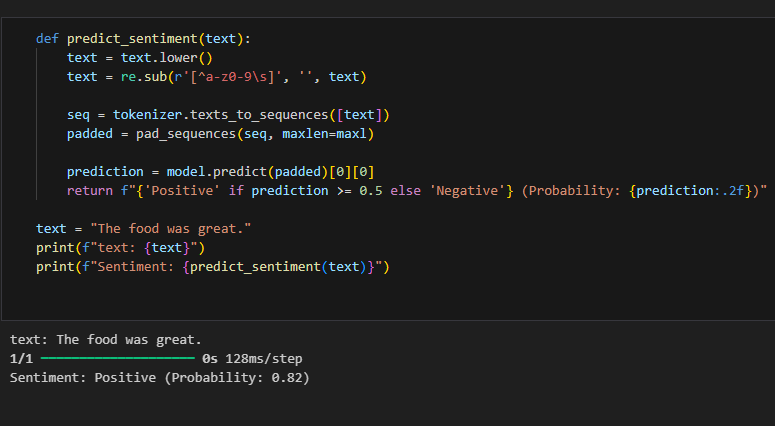


3. Build a Recurrent Neural Network model.

  
4. Train the model using binary crossentropy.

  
5. Evaluate the model on the test set.

6. Make predictions with probability scores.



# Model Output and Accuracy

The trained RNN model was tested on a separate validation set and achieved high performance. For example, for the input: "The food was great", the model predicted a \*\*positive\*\* sentiment with a \*\*probability of 0.82 (82%)\*\*.

# Conclusion

The RNN-based sentiment analysis model successfully classified text inputs with good accuracy. With a prediction confidence of 82% on test samples, this model shows potential for use in customer feedback systems, social media analysis, and other natural language processing tasks.