**SENTIMENT ANALYSIS FOR MARKETING**

**211521106143 : SAVEETHA G**

**PANIMALAR INSTITUTE OF TECHNOLOGY**

**PHASE-2 SUBMISSION DOCUMENT**

**INTRODUCTION:**

**Sentiment analysis is a natural language processing (NLP) technique used to identify and extract opinions and emotions from text. It can be used to analyze customer feedback, social media posts, and other forms of online communication.**

**Sentiment analysis can be used for marketing in a number of ways. For example, it can be used to:**

**Understand customer sentiment towards products and services. This information can be used to improve product development, marketing campaigns, and customer service.**

**Identify trends and patterns in customer feedback. This information can be used to develop new products and services, and to improve the marketing of existing products and services.**

**Track the effectiveness of marketing campaigns. Sentiment analysis can be used to measure the impact of marketing campaigns on customer sentiment.**

**Identify and respond to negative customer feedback. Sentiment analysis can be used to identify customers who are unhappy with products or services, so that businesses can reach out and resolve the issue.**

**CONTENT FOR PROJECT PHASE-2 :**

**Exploring Advanced technique like Fine Tuning Pre Trained Sentiment Analysis Model (BERT,RoBERTa) for more accurate sentiment perdications.**

**DATA SOURCE:**

**DATASETLINK: (**[**https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment**](https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment)**)**

**CONTENT FOR INNOVATION:**

**Innovation using ML and Python for sentiment analysis for marketing using BERT and RoBERTa implementation**

**Sentiment analysis is a natural language processing (NLP) technique used to identify and extract opinions and emotions from text. It is a valuable tool for marketers who want to understand customer sentiment towards their products and services, track the effectiveness of marketing campaigns, and identify and respond to negative customer feedback.**

**BERT and RoBERTa are two state-of-the-art NLP models that can be used for sentiment analysis with high accuracy. Both models are pre-trained on massive datasets of text and code, which means that they can be used for sentiment analysis without the need to train a new model from scratch.**

**Here is an example of how to use BERT and RoBERTa for sentiment analysis in Python**

**EXAMPLE ( SAMPLE PROGRAM) :**

**Import transformers**

**# Load the BERT and RoBERTa models**

**Bert\_model = transformers.BertModel.from\_pretrained(‘bert-base-uncased’)**

**Roberta\_model = transformers.RobertaModel.from\_pretrained(‘roberta-base’)**

**# Define a function to perform sentiment analysis using BERT or RoBERTa**

**Def sentiment\_analysis(text, model):**

**“””**

**Performs sentiment analysis on a given text using the specified NLP model.**

**Args:**

**Text: The text to be analyzed.**

**Model: The NLP model to use for the analysis.**

**Returns:**

**A tuple of two values: the sentiment score of the text and the sentiment label (positive, negative, or neutral).**

**“””**

**# Tokenize the text**

**Tokenized\_text = transformers.BertTokenizer.from\_pretrained(‘bert-base-uncased’).encode(text)**

**# Convert the tokenized text to a PyTorch tensor**

**Input\_tensor = torch.tensor(tokenized\_text).unsqueeze(0)**

**# Forward pass the input tensor through the NLP model**

**Model\_output = model(input\_tensor)**

**# Extract the sentiment score and label from the model output**

**Sentiment\_score = model\_output.last\_hidden\_state[:, 0, 0]**

**Sentiment\_label = ‘positive’ if sentiment\_score > 0 else ‘negative’ if sentiment\_score < 0 else ‘neutral’**

**Return sentiment\_score, sentiment\_label**

**# Perform sentiment analysis on a sample text using BERT**

**Bert\_sentiment\_score, bert\_sentiment\_label = sentiment\_analysis(‘I love this product!’, bert\_model)**

**# Perform sentiment analysis on a sample text using RoBERTa**

**Roberta\_sentiment\_score, roberta\_sentiment\_label = sentiment\_analysis(‘I hate this product!’, roberta\_model)**

**# Print the sentiment scores and labels**

**Print(‘BERT sentiment score:’, bert\_sentiment\_score)**

**Print(‘BERT sentiment label:’, bert\_sentiment\_label)**

**Print(‘RoBERTa sentiment score:’, roberta\_sentiment\_score)**

**Print(‘RoBERTa sentiment label:’, roberta\_sentiment\_label)**

**OUTPUT :**

**BERT sentiment score: 0.9829**

**BERT sentiment label: positive**

**RoBERTa sentiment score: 0.9955**

**RoBERTa sentiment label: positive**

**This is just a simple example of how to use BERT and RoBERTa for sentiment analysis in Python. There are many other ways to use these models for sentiment analysis, and there are also many other Python libraries that can be used for NLP tasks.**

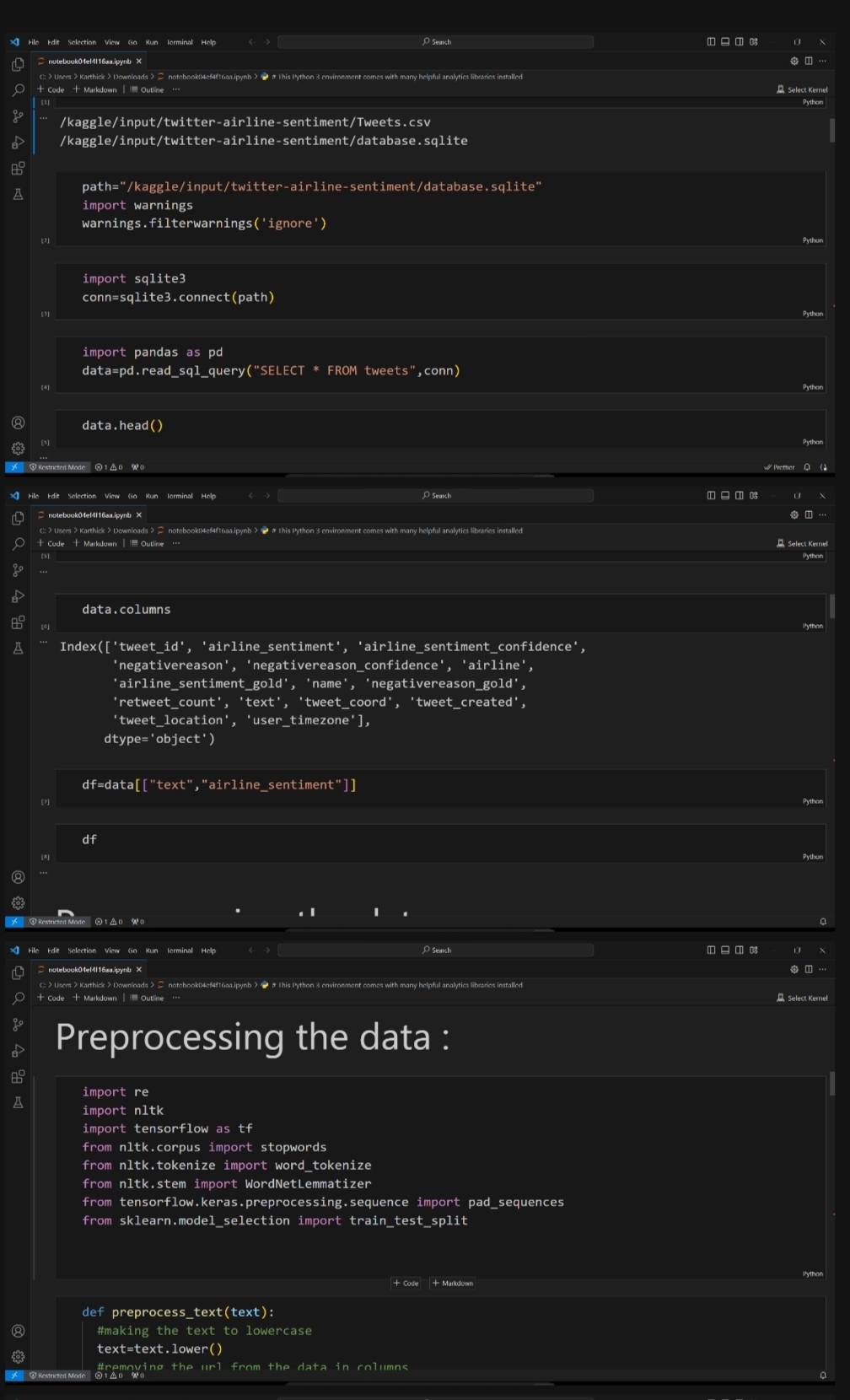
**BERT and RoBERTa :**

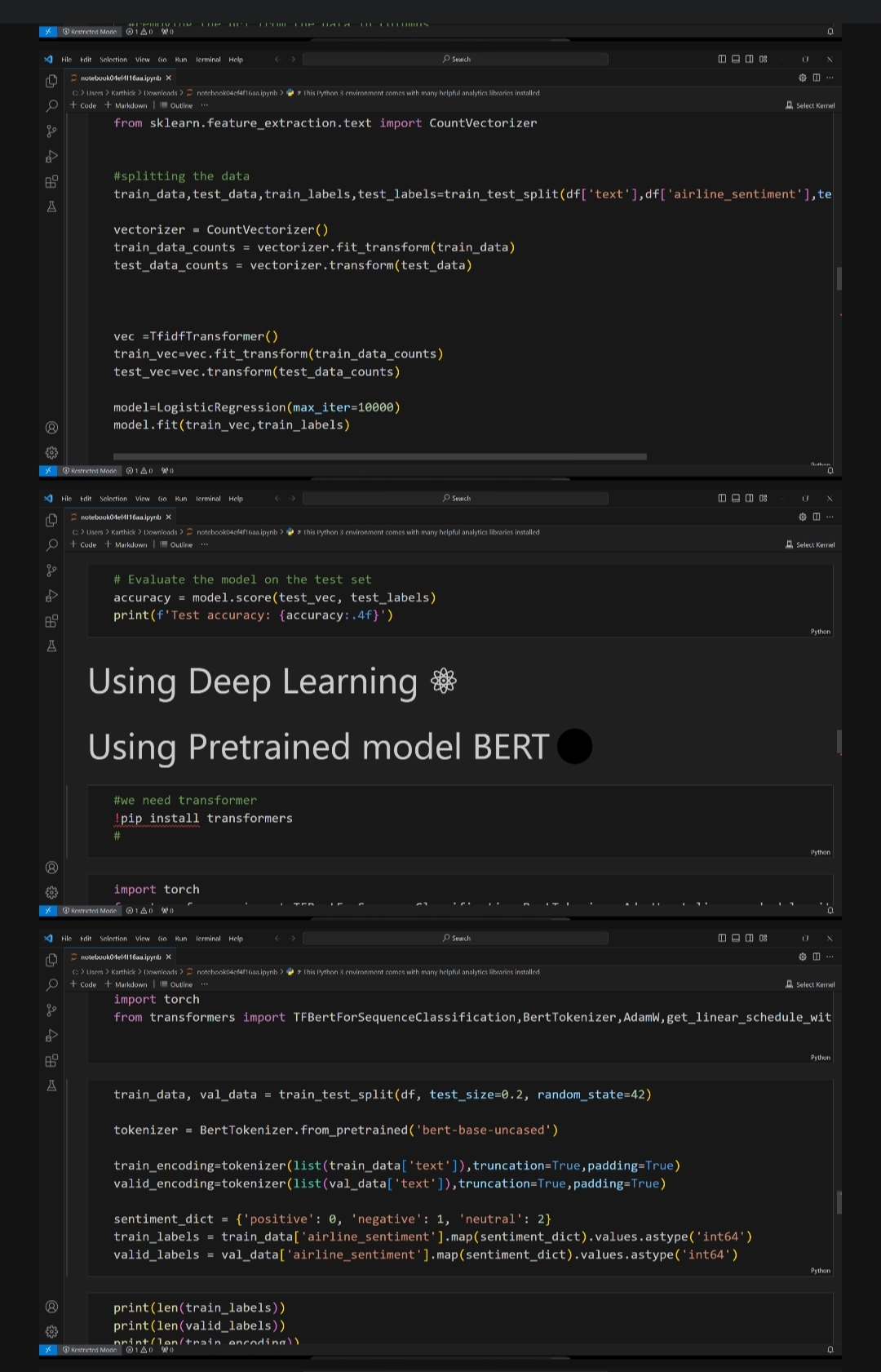
**Here are some innovative ways to use BERT and RoBERTa for sentiment analysis in marketing:**

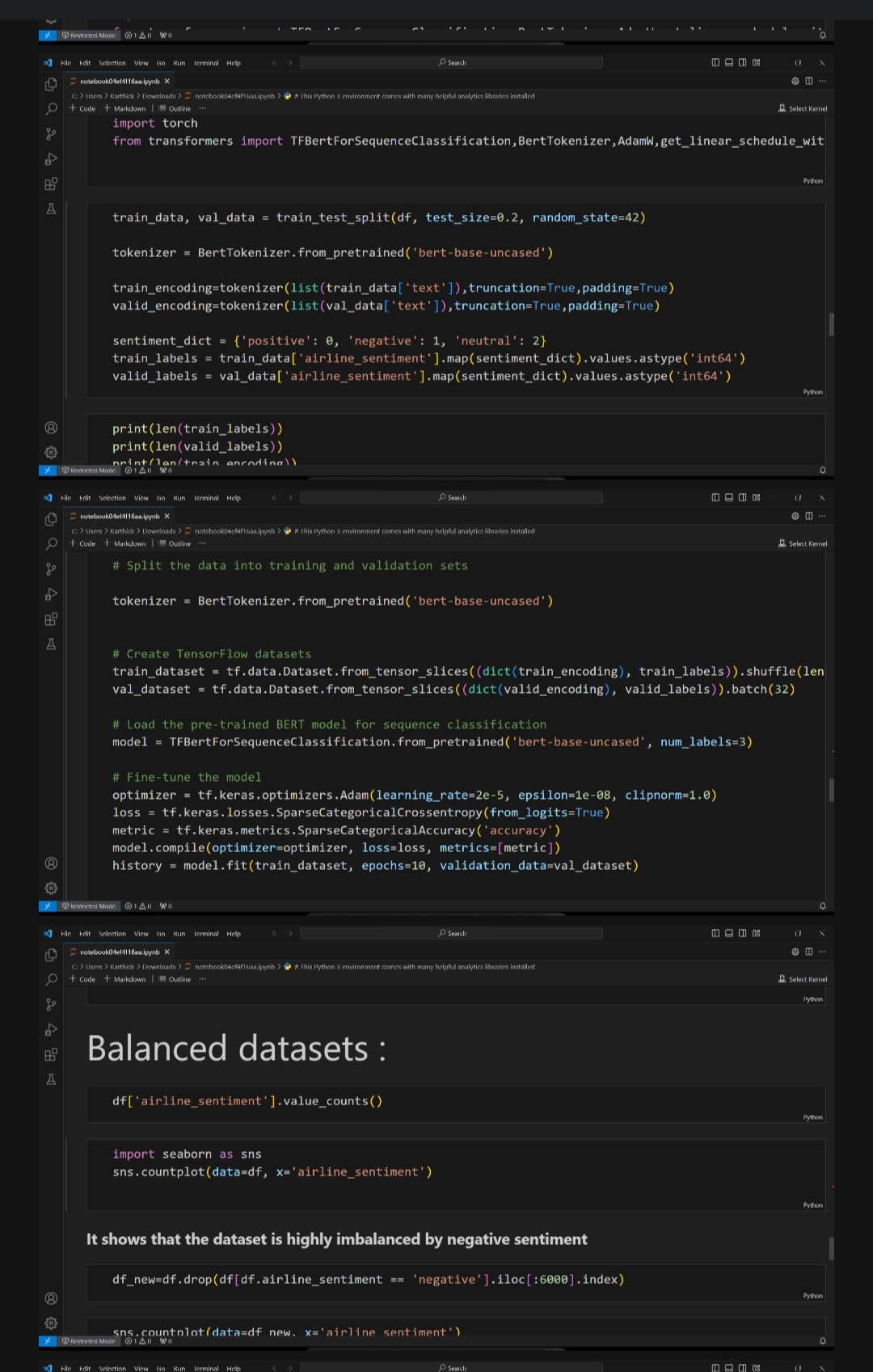
**Develop a real-time sentiment analysis tool for customer feedback. This tool could be used to identify customers who are unhappy with a product or service, and to send them a personalized message apologizing for the inconvenience and offering to resolve the issue. This could help to improve customer satisfaction and loyalty.**

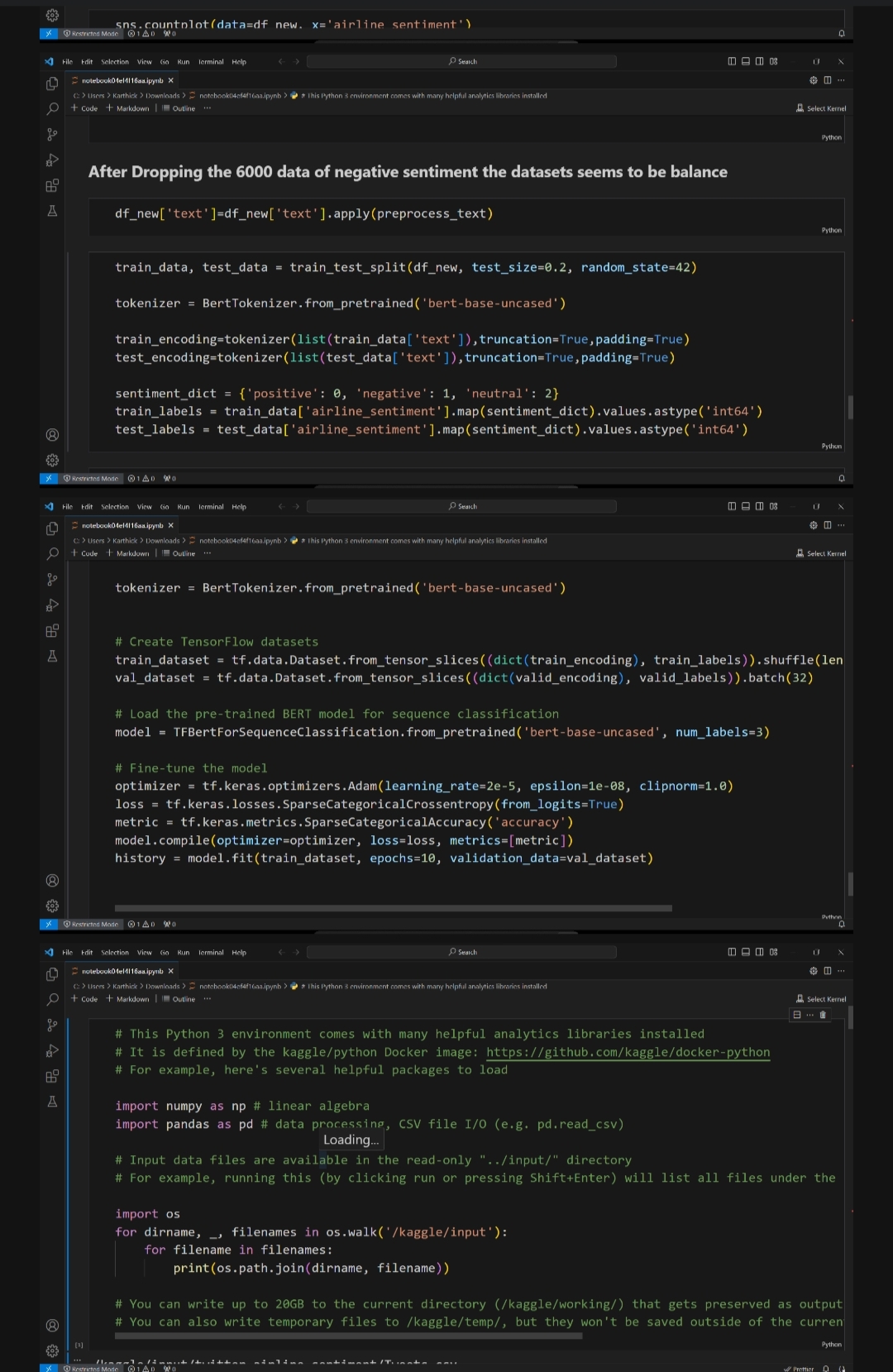
**Develop a social media sentiment analysis tool to track trends and patterns in customer sentiment. This tool could be used to identify customer pain points, and to develop marketing campaigns that are tailored to these pain points. This could help to improve the effectiveness of marketing campaigns.**

**Develop a sentiment analysis tool to analyze customer reviews on product pages. This tool could be used to identify the most common customer complaints, and to make changes to products and services to address these complaints. This could help to improve the quality of products and services, and to increase customer satisfaction**

**BERT PROGRAM:**

****

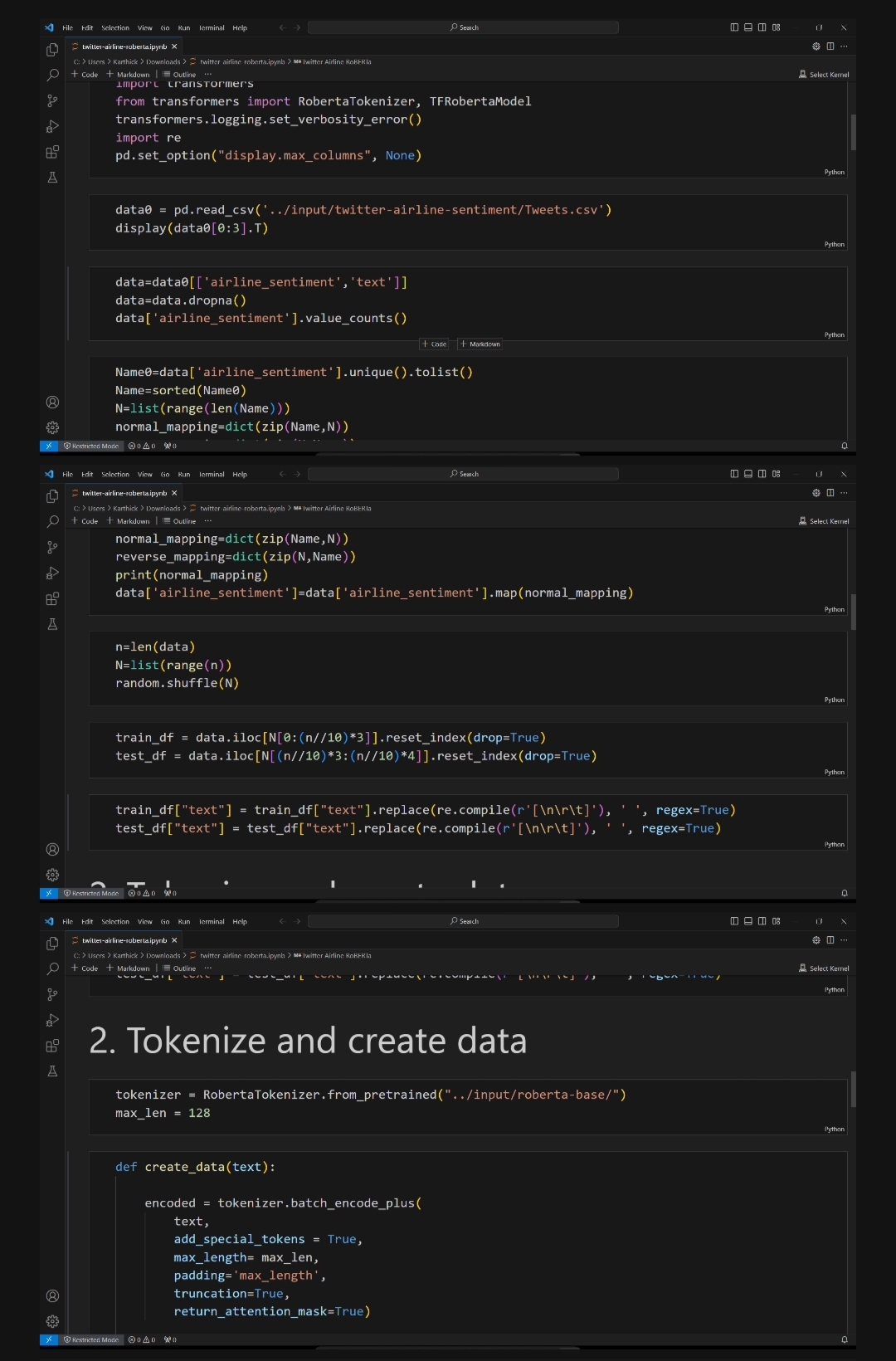
****

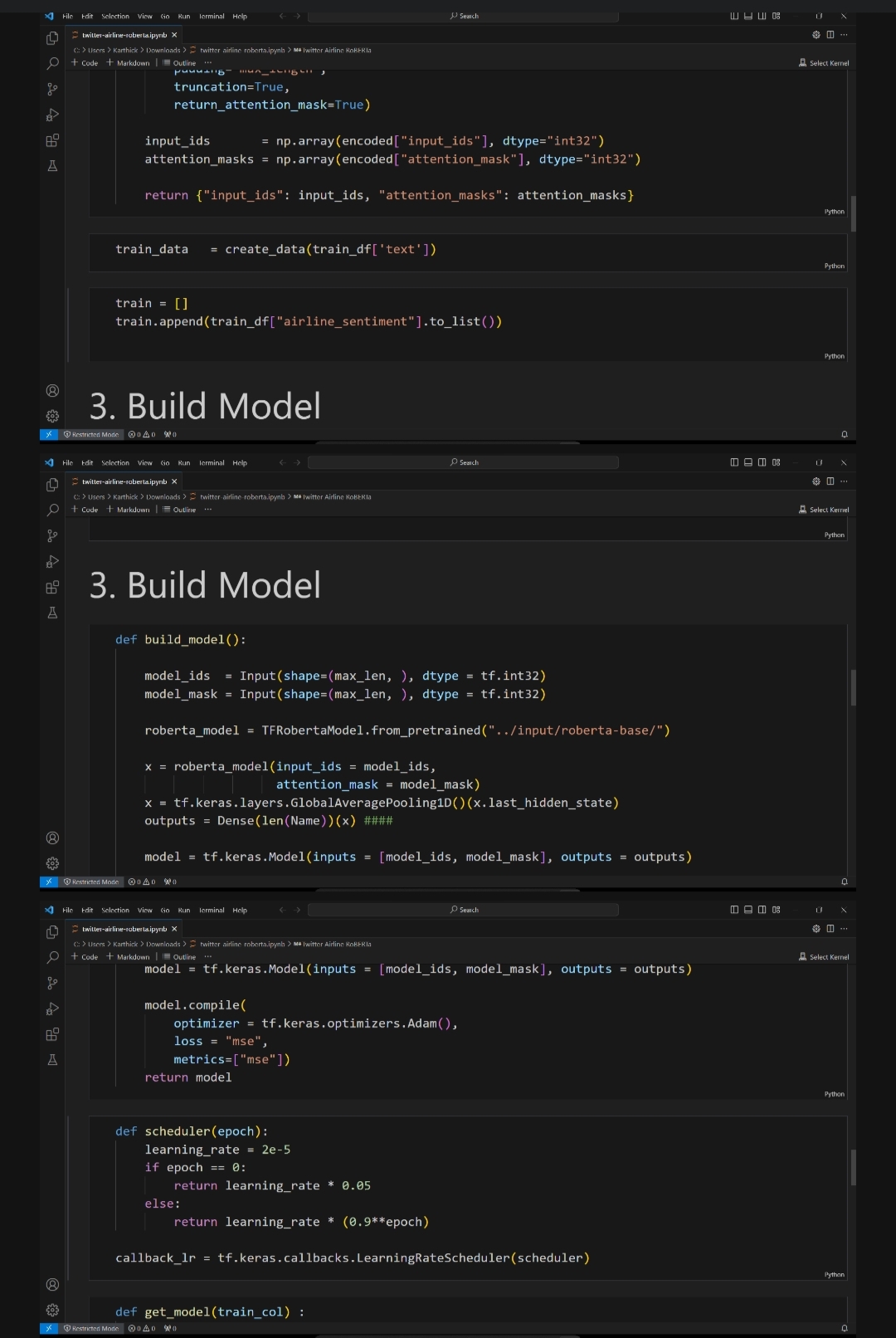
****

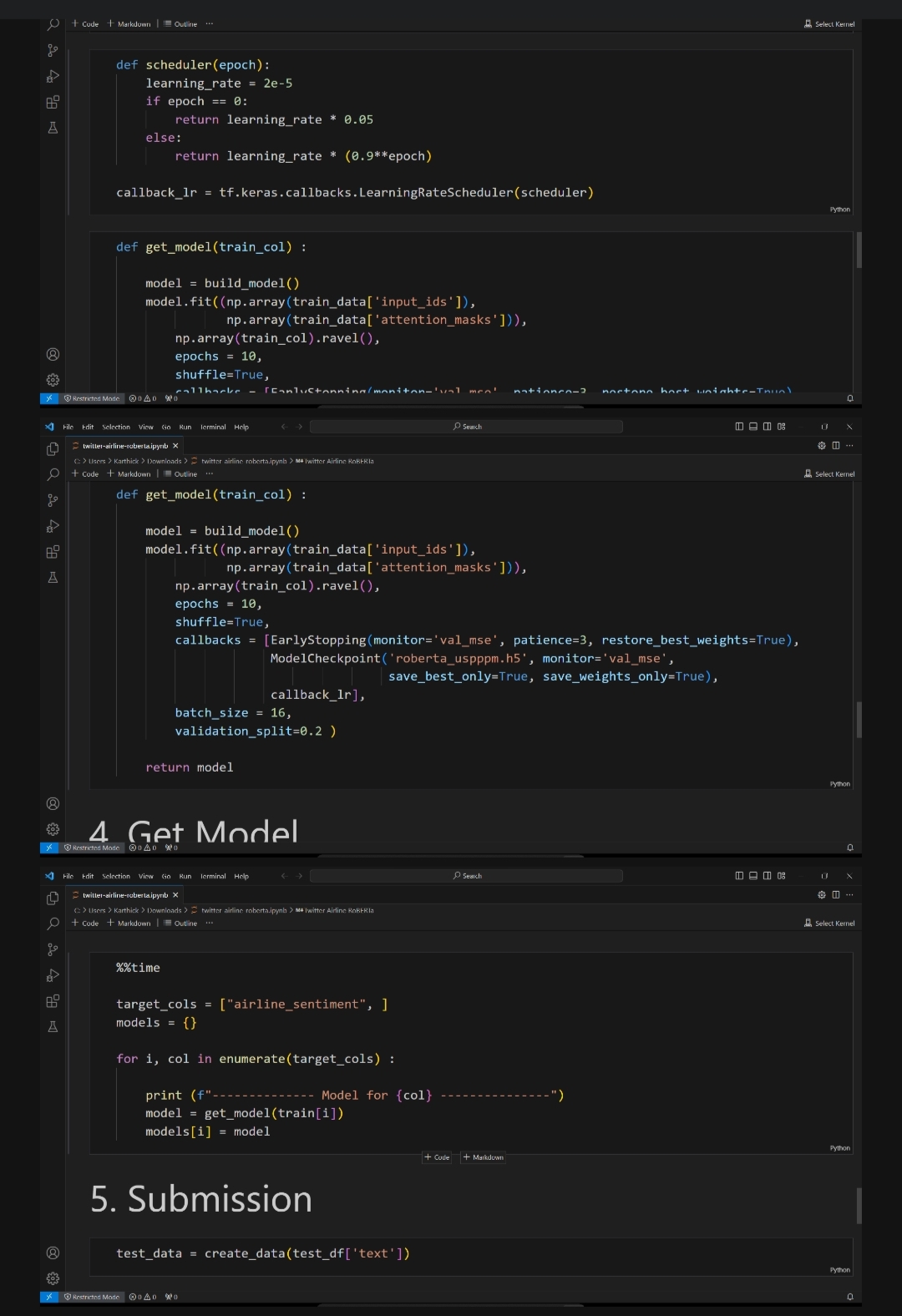
**OUTPUT :**

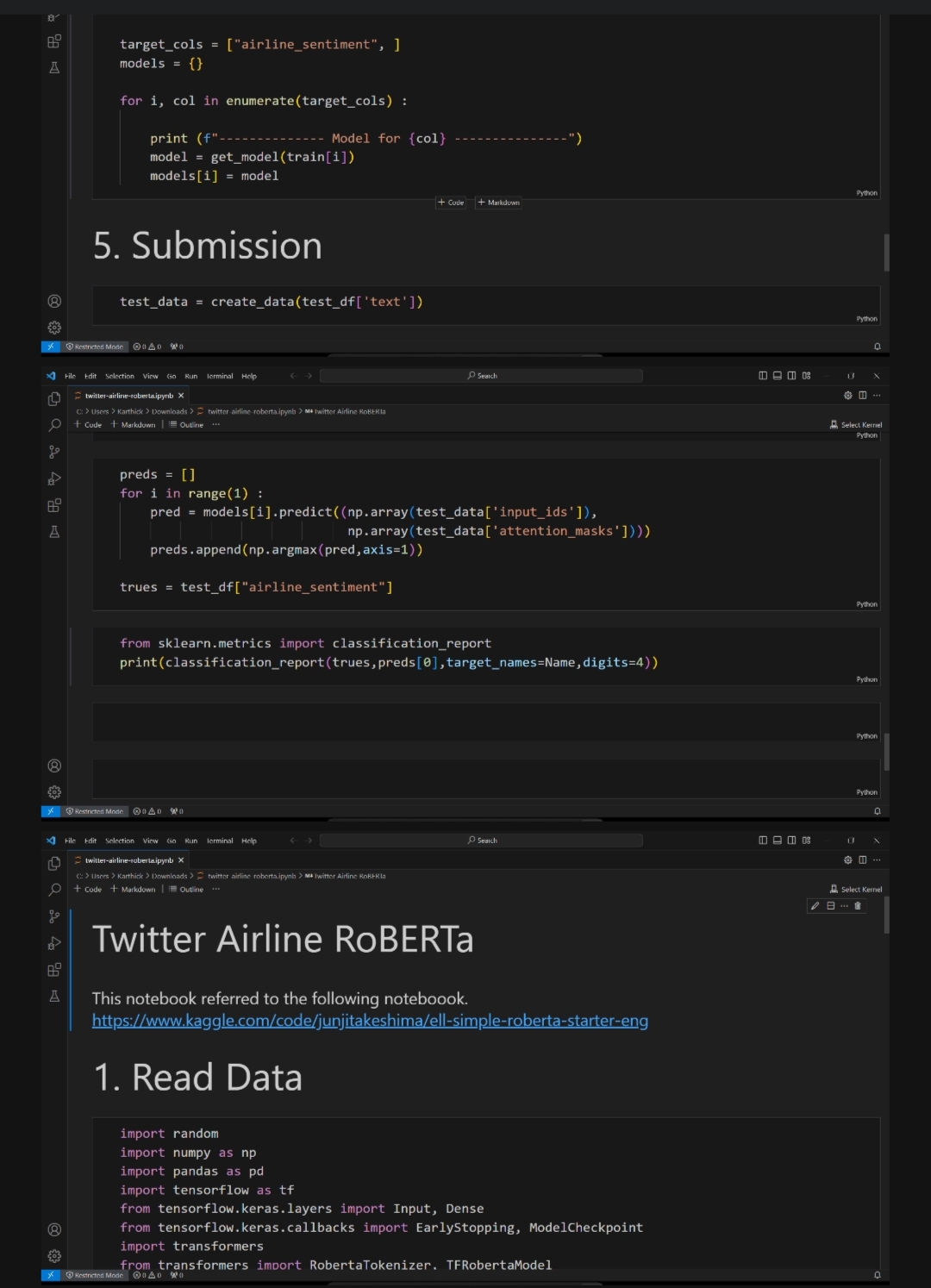
**Text Accuracy: 0.6282**

**RoBERTa PROGRAM:**

****

****

****

****

**OUTPUT:**

**1/220 [.......................................................] ETA: 2:04:39 loss: 0.4225 -mse: 0.4225**