**Airline Sentiment Analysis Tool: Analyzing and Classifying Airline Tweets**

**Description:**

The “Airline Sentiment Analysis Tool” is a robust application designed to analyze and classify tweets related to various airlines. As a programmer, your task is to develop this application with specific requirements in mind.

**Key Features:**

* First, the application will **check** whether there is “**model.pickle**” file in the application directory or not.
* If the file **exists**, then the application will **read** and **load** the **data training** from the **file**.
* If the file **doesn’t exist**, then the application will **train** the tweet data with **Naïve Bayes Classifier** from **NLTK data twitter sample** provided in “**dataset.csv**”. The **data training** will be following these rules:
* **Preprocess** the dataset by **tokenizing the words**, **remove stopwords**, **remove symbols and number**, **stemming**, and **lemmatizing the words**.
* Compare the tweet words with the words in list of dictionaries.
* If the tweet is in the **positive** category, then set the **tweet category** to **positive**.
* If the tweet is in the **negative** category, then set the **tweet category** to **negative**.
* **Train** the model using **Naïve Bayes**.
* **Show 5 most informative features** and **training accuracy**.
* **Save training model to pickle file** with format name “**model.pickle**”.

Text

Description automatically generated with low confidence

* The application menu will **show user tweets** and consist of **3 menus**. **Validate** in the menu, that user can only choose number in the range of menu provided (**1 - 3**).
  + - 1. **Write tweet**
      2. **Analyze tweet**
      3. **Exit**
* If user choose **menu 1** (“**Write tweet**”), then the application will:
* Ask the user to input tweet. **Validate** that the input must **at least contains of 5 words**.
* After that, the application will **save** the **tweet**.
* If user choose **menu 2** (“**Analyze tweet**”), then the application will:
* **Check** whether there is a **tweet** or not.
* If there is no **tweet exists**, then show a **message** to notify the user and redirect user **back to main menu**.
* If there is **tweet exists**, then the application will do these following procedures:
* **Show Part of Speech** (**POS**) **Tagging**.

**Background pattern

Description automatically generated with low confidence**

* **Show** the **synonyms** and **antonyms** of the word in the tweet. If the word doesn’t have any **synonym** or **antonym**, **show message** to notify the user.

A picture containing chart

Description automatically generated

* **Predict** and show the result of the **tweet category**.

****

* If user choose **menu 3** (“**Exit**”), then **terminate** the application.