FACULTY OF ENGINEERING AND ARCHITECTURE

ARTIFICIAL INTELLIGENCE APPLICATIONS

Kübra BAHADIR

1904010014

CAPSTONE PROJECT

Emotions Detection in Turkish Text

1. Introduction

This project focuses on detecting emotions from text data using machine learning techniques. The goal is to build a model that can accurately classify the emotion expressed in a given text. This project specifically uses Streamlit to create a web application that performs sentiment analysis on Turkish text inputs. The application takes Turkish text as input, translates it to English using the Google Translate API, and then uses a pre-trained machine learning model to predict the emotion expressed in the text. The application displays the predicted emotion along with an appropriate emoji and the prediction probability. Additionally, the application uses an SQLite database to keep track of page visits and prediction details.

2. Libraries and Packages

The following libraries and packages are used in this project:

- **Data manipulation and visualization**: `pandas`, `numpy`, `seaborn`
- **Text cleaning**: `neattext.functions`
- **Machine learning**: `LogisticRegression`, `MultinomialNB`, `CountVectorizer`, `train_test_split`, `accuracy_score`, `classification_report`, `confusion_matrix`
- **Web application**: `streamlit`
- **Translation**: `googletrans`
- **Database**: `sqlite3`
- **Other**: `joblib`, `altair`, `plotly`, `pytz`

3. Data Loading

The dataset used for this project is loaded from a CSV file called `emotion_dataset_raw.csv`. The data is loaded into a pandas DataFrame `df`, and the first few rows are displayed using `df.head()`.

4. Exploratory Data Analysis (EDA)

Exploratory data analysis is performed to gain insights into the dataset. The value counts of emotions are calculated, and a count plot is created to visualize the distribution of emotions.

5. Data Cleaning

The text data is cleaned by removing user handles and stopwords using the `neattext.functions` library.

6. Feature Extraction and Labeling

The cleaned text is used as the feature, and the corresponding emotions are used as labels. These features and labels are extracted from the DataFrame.

7. Data Splitting

The dataset is split into training and testing sets using the `train_test_split` function from `sklearn`.

8. Model Building

A pipeline is created with `CountVectorizer` for feature extraction and `LogisticRegression` for classification. The model is trained on the training data using `pipe_lr.fit(x_train, y_train)`.

9. Model Evaluation

The performance of the trained model is evaluated on the test data. The accuracy score is calculated using `pipe_lr.score(x_test, y_test)`. Additionally, predictions are made on the test data, and their probabilities are checked.

10. Model Saving

The trained model and pipeline are saved using `joblib` for future use and deployment.

11. Web Application

The project also includes a Streamlit web application that allows users to input Turkish text and analyze the emotions expressed in the text. The application consists of the following files:

1. **app2.py**: This file contains the main code for the Streamlit web application. It creates the user interface, takes text input, performs sentiment analysis, and displays the results.
2. **track_utils.py**: This file contains helper functions for performing database operations. It records page visits and prediction details, and provides functions to view this data.
3. **emotion_classifier_pipe_lr.pkl**: This file contains the pre-trained machine learning model used by the application to predict the emotion expressed in the input text.
4. **data.db**: This file is an SQLite database used to store page visit and prediction details.

The web application uses various Python libraries, including Streamlit, Google Translate API, TextBlob, Altair, Plotly, Pandas, NumPy, SQLite, and pytz. The purpose of the project is to allow users to analyze the emotions expressed in Turkish text and visualize the prediction results.

This notebook provides a comprehensive workflow for emotion detection in text, from data preprocessing to model deployment, and includes the details of the Streamlit web application for sentiment analysis on Turkish text.