

**Mainstream Natural language processing**

# **Sentiment Analysis of movie reviews**

# Team T089

### Team Members:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Student Name | Official E-mail | Level | Department |
| 20201700113 | **اسلام احمد محمد اسماعيل** | 20201700113@cis.asu.edu.eg | 3 | **cs** |
| 20201700082 | **احمد مدحت فوزي منجود** | 20201700082@cis.asu.edu.eg | 3 | **cs** |
| 20201700301 | **زياد اسلام حمدي عبد الغفار** | 20201700301@cis.asu.edu.eg | 3 | **cs** |
| 20201700519 | **عمر محمد محمود اسماعيل** | 20201700519@cis.asu.edu.eg | 3 | **cs** |
| 20201701226 | **ميرا ميخائيل سمير اسحق** | 20201701226@cis.asu.edu.eg | 3 | **cs** |
| 20201700587 | **فؤاد احمد سامي فؤاد الدهان** | 20201700587@cis.asu.edu.eg | 3 | **cs** |

**1-Data Reading:**

-we use function ”read data” that reads all text files from multiple folders, assigns the folder name as the label for each file, and returns a pandas DataFrame with the text content and label for each file.

**2-Data Preprocessing:**

**-** the program reads text data from multiple folders, preprocesses the data by converting it to lowercase, removing punctuations and stopwords, stemming by **(SnowballStemmer)**, and lemmatizing words by **(WordNetLemmatizer)**. It then splits the data into training and validation sets and encodes the labels.

**3-Features Extraction:**

- The program creates a TF-IDF vectorizer object, fits it on the training data, and transforms both the training and validation sets using the vectorizer.

**4-Model Training and Testing:**

-The code trains several classifiers, including **Naive Bayes**, **SVM**, **Random Forest**, **Gradient Boosting**, and **Decision Tree**, on the transformed data using the **“train\_model”** function. It prints the accuracy scores of each classifier and identifies the best-performing model using the **“print\_best\_model”** function.

**5-Results Visualization:**

-The code creates a bar chart to visualize the training and testing accuracies of each classifier. It loops through the classifiers, trains them on the transformed training data, and calculates their training and testing accuracies. It then creates a bar chart with the model names on the x-axis and the accuracies on the y-axis. The training accuracies are plotted in blue and the testing accuracies are plotted in red. The bar chart is displayed using the **“plt.show()”** function.

- The code plots the confusion matrix, it uses the best-performing classifier**(svm)** to predict the labels for the validation set, calculates the confusion matrix using **scikit-learn's confusion\_matrix function**, and plots the confusion matrix using the **plot\_confusion\_matrix** function

**6-Screen Shots From The Run:**

* Accuracy of each classifier:

![A screen shot of a computer

Description automatically generated with medium confidence]()

* model training and testing chart:

![A picture containing text, screenshot, rectangle

Description automatically generated]()

* The Confusion Matrix of the svm classifier:

![A picture containing text, rectangle, software, diagram

Description automatically generated]()