SENTIMENT ANALYSIS FOR MOVIE RECOMMENDATIONS

By Hilario Jr Nengare

Advisor Prof. Dr. Zafer Erenel

INTRODUCTION: Sentiment Analysis

STEPS OF SENTIMENT ANALYSIS

DATA COLLECTION

TEXT PREPARATION

SENTIMENT DETECTION

SENTIMENT CLASSIFICATION

DATA ANALYSIS

DATA LOADING

Loaded data set version 1 csv file comprised of the srcaped user reviews obtained from the IMDB website.

INFO ON DATA

General information about the data set was acquired using the pandas.info and pandas.describe methods.

DROPPING

The duplicates and missing tabs were dropped by invoking the pandas.drop method.

DATA TOKENIZATION

DATA IS LOADED

Here data saved from the analysis is loaded,

PUNKT

Downloaded the PUNKT library, a necessity for tokenization.

WORD TOKENIZE

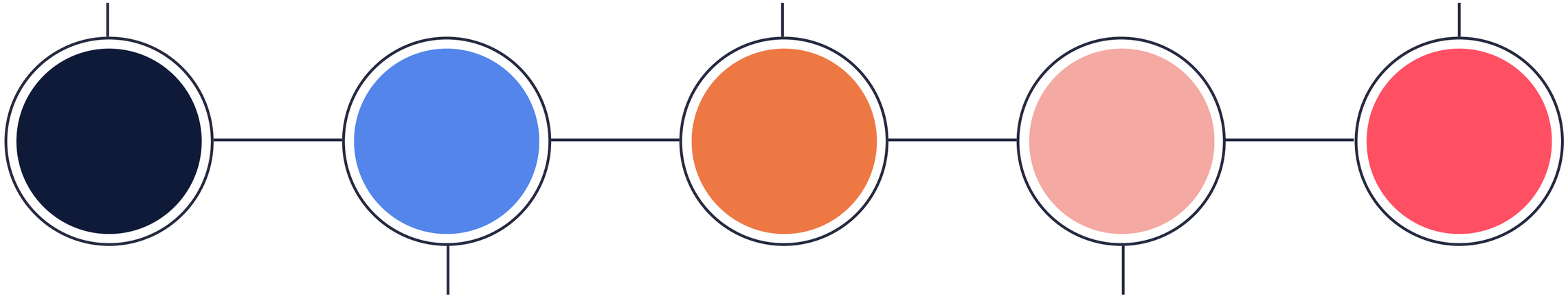
Used the NTLK ‘s word_tokenize method to transform the data set version 2 into a list of words.

STOP WORDS REMOVAL

Downloaded the NLTK stop_words library and created a set of stop words which then I filtered out.

PICKLE

The filtered file is pickled.



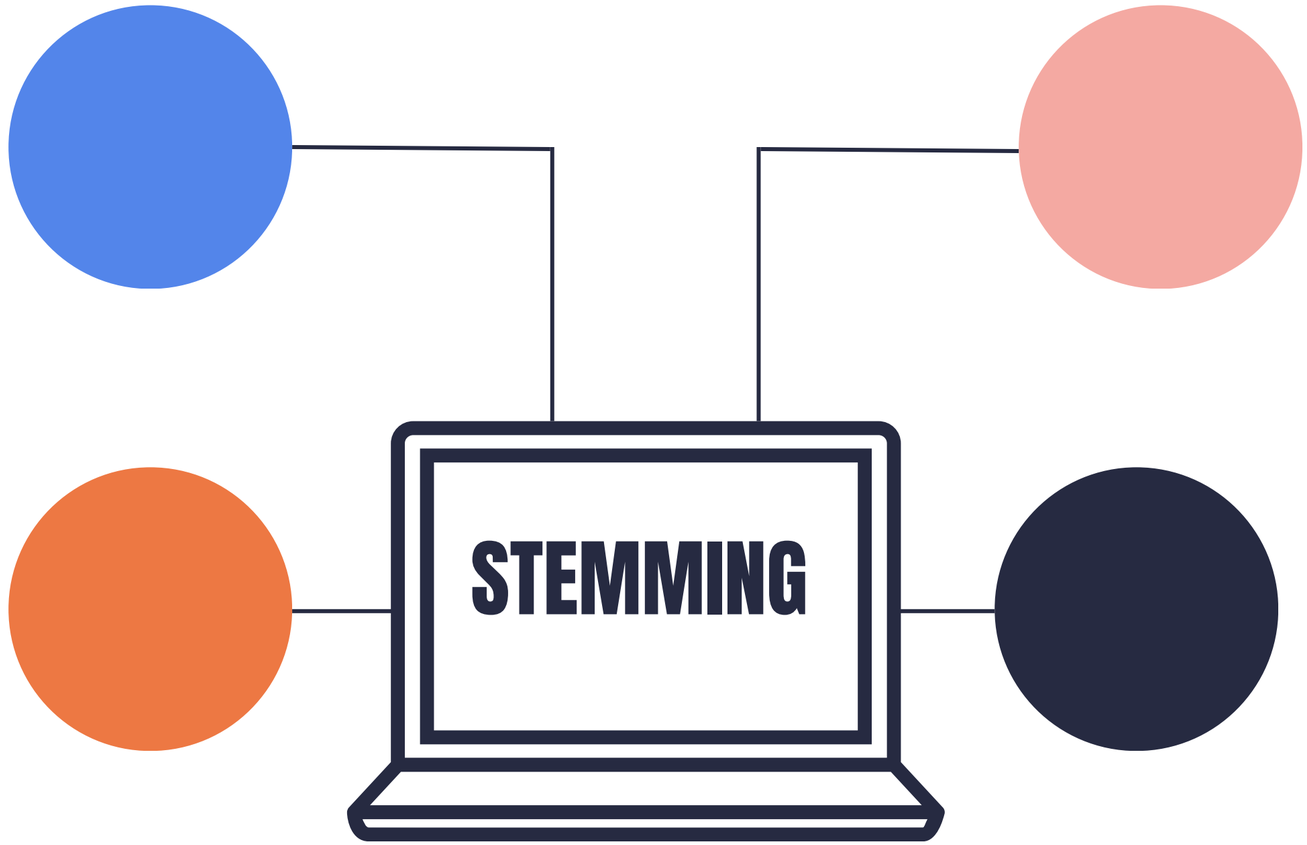
DATA STEMMING

DATA IS LOADED

The tokenized data set version 3 pickle file is loaded .

PORTERSTEMMER

used NLTK’s Porter Stemmer to reduce words to their root or base form.



DATA REGEXP

Filtered out all the unnecessary symbols before stemming using regular expressions.

PICKLE FILE

Saved the file in pickle format.

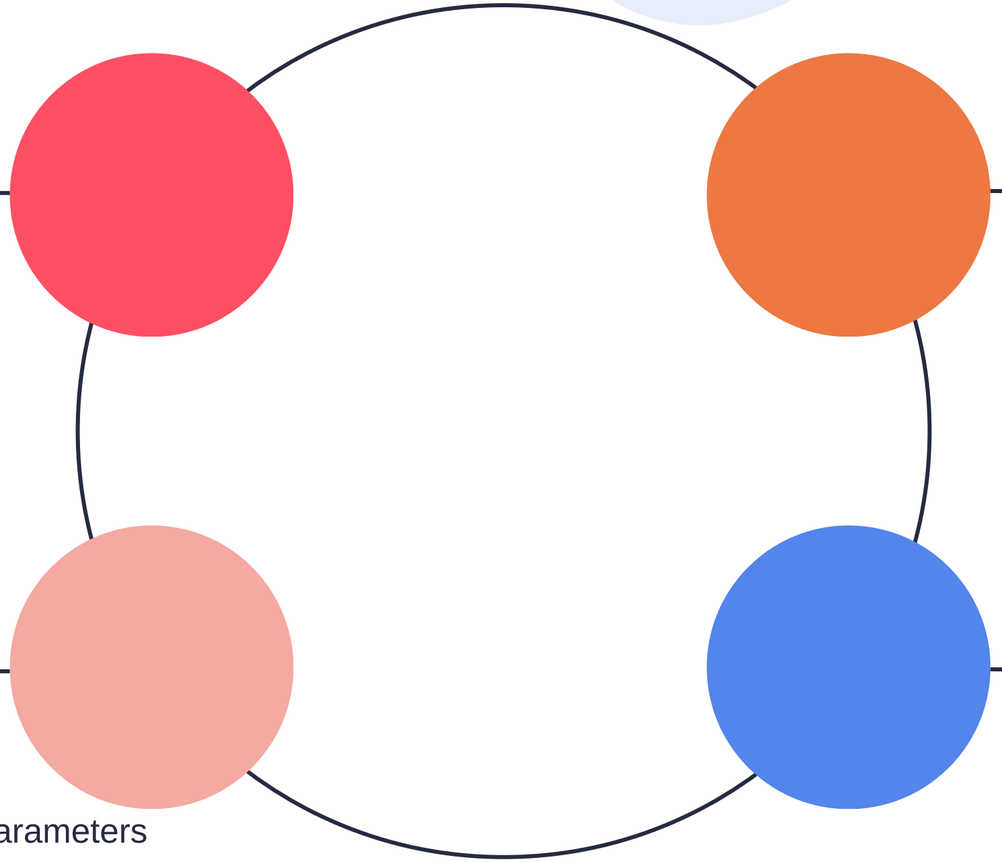
MODEL TRAINING

DATA VECTORIZATION

Preprocessed the movie reviews dataset, split it into training and testing sets, and applied TF-IDF vectorization to convert the text data into numerical format suitable for model training.

HYPERPARAMETER TUNING

Performed Hyperparameter tuning using GridSearchCV in order to find the optimal parameters for the SVM model, leading to improved performance on the test data.



TRAINING AND EVALUATION

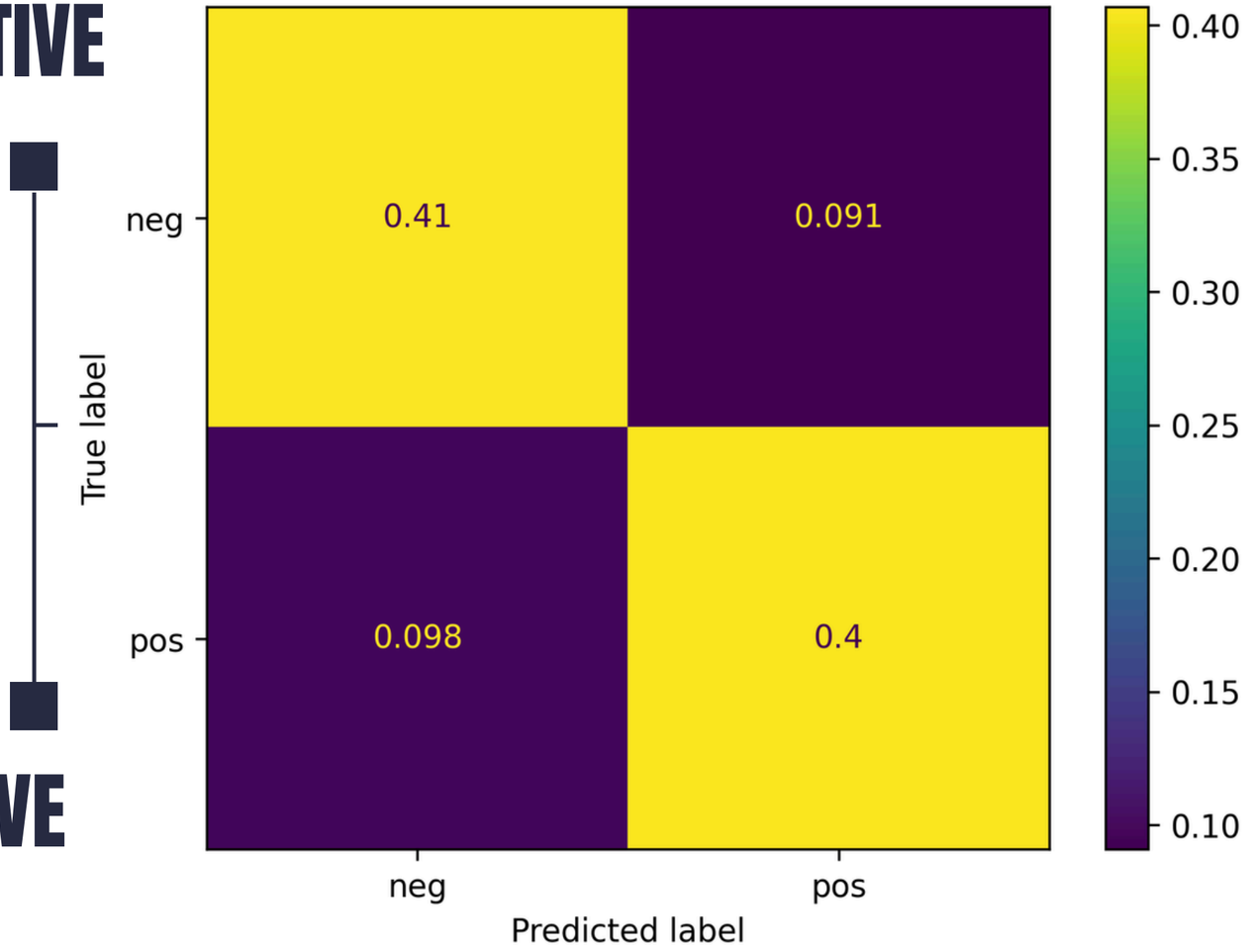
Trained a Support Vector Machine (SVM) classifier on the training data, snd evaluated the performance using accuracy and F1 score metrics, and displayed the confusion matrix for further insights.

MODEL INTERPRETATION

Used LIME, to interpret the SVM classifier's predictions for a sample review and saved the optimized model and the entire pipeline to disk for usage in the Flask server.

CONFUSION MATRIX

NEGATIVE



POSITIVE

FLASK APPLICATION

FLASK CONFIGURATION

- The Flask app is configured with a secret key and session type.
- I set the template and static folder paths.
- Flask-Session is used to manage server-side sessions

MODEL LOADING

- The SVM model pipeline is loaded from the pickle file for performing sentiment analysis.
- Movie data is fetched from the OMDb API things asndetailed plots, using the load_movies and load_movie functions to retrieve and parse movie information based on queries from users.

CONTENT-BASED FILTERING

- The **ContentBasedFilter** class is defined to recommend movies based on genre, plot, director and actors.
- Used TF-IDF vectorization and cosine similarity to calculate movie similarities and recommend similar movies based on the provided IMDb ID.

REVIEW FORM

- Used the FlaskForm class to define the ReviewForm, with a text area for user reviews and a submit button, incorporating data validation.

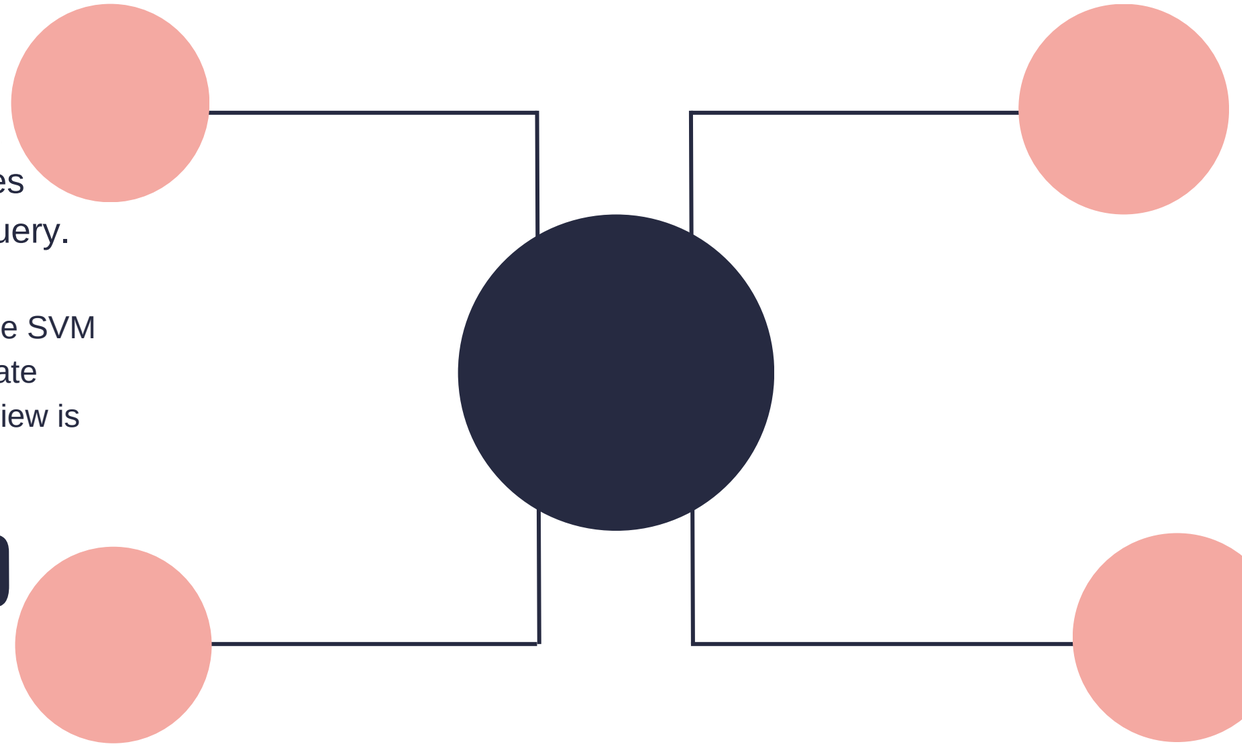
ROUTES AND RECOMMENDATIONS

INDEX ROUTE

- Displays the main page with a review form and a list of movies based on a genre or search query.
- Handles form submission, uses the SVM model to classify reviews and initiate movie recommendations if the review is positive.

RECOMMENDED MOVIES ROUTE

- This page displays recommended movies stored in the session.
- Renders those recommendations using the recommended.html template.



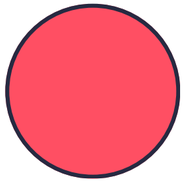
REVIEW SUBMISSION

- When a positive review is submitted, ContentBasedFilter is used to recommend similar movies.
- Recommended movies are stored in the session and the user is redirected to a recommendations page with a flash message informing them its a positive review.

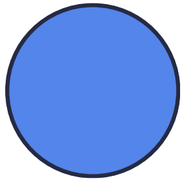
UTILITY FUNCTIONS

- The load_movies and load_movie functions handle API requests to fetch movie data.
- These functions ensure movie data retrieval,..

CONCLUSION: Benefits of SA



Sentiment analysis helps in understanding not just what items users interact with, but how they feel about them.



Understanding the sentiment behind user reviews and comments, recommender systems can provide more contextually relevant suggestions.

Recommendations that align closely with user sentiments, the overall satisfaction and engagement with the platform can be increased.