Abdelmalek Essaadi University
Faculty of Science and Technology Tangier
Computer Engineering Department
Master: AIDS
Realized by : SABBAHI Mohamed Amine
Framed by : Pr . ELAACHAk LOTFI

Synthesis of lab 3 (NLP)

Introduction:

In this lab, I explored various Natural Language Processing (NLP) techniques and machine learning models with Sklearn to analyze text data, with a focus on language modeling and sentiment analysis using different word embedding methods. My primary objective was to gain insights into how these techniques can be applied to real-world datasets to extract meaningful information and make accurate predictions. Here's a synthesis of my key learnings:

1. NLP Preprocessing Pipeline:

The first step in our NLP pipeline involved preprocessing the text data. This included:

- **Tokenization:** Splitting text into individual words or tokens.
- **Stemming and Lemmatization:** Reducing words to their root form.
- **Stop Words Removal:** Removing common words that do not carry significant meaning.
- **Discretization:** Converting continuous data into discrete buckets, although not commonly used in text preprocessing.

These preprocessing steps are essential for cleaning the text data and preparing it for further analysis.

2. Word Embedding Techniques:

I employed several word embedding techniques to represent the text data in numerical form:

- **Bag of Words (BoW):** Representing text by counting the frequency of each word in a document.
- **TF-IDF (Term Frequency-Inverse Document Frequency**): Assigning weights to words based on their frequency in a document relative to their frequency across all documents.
- **Word2Vec (CBOW and Skip Gram):** Using neural networks to learn word embeddings by predicting the context of words (CBOW) or the words given a context (Skip Gram).

These embeddings are crucial for transforming text data into a format that machine learning models can work with.

3. Machine Learning Models:

I trained several machine learning models on the preprocessed and embedded text data:

Regression Models (for language modeling):

- SVR (Support Vector Regression)

- Linear Regression
- Decision Tree Regressor

Classification Models (for sentiment analysis):

- SVM (Support Vector Machine)
- Logistic Regression
- Ada Boosting

4. Model Evaluation:

I evaluated the models using various performance metrics:

Regression Models: Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).

Classification Models: Accuracy, F1 Score, Precision, Recall, and additional metrics like BLEU score.

The evaluation metrics helped me determine the effectiveness of each model and select the best one for our tasks.

5. Interpretation of Results:

From ym evaluations, I found that:

- **Linear Regression with CBOW embeddings** was the best model for language modeling, providing the lowest MSE and RMSE.
- **SVM with Skip Gram embeddings** was the best model for sentiment analysis, achieving the highest accuracy and F1 Score.

These results highlight the importance of choosing the right embedding technique and machine learning model for specific NLP tasks.

Conclusion:

This lab provided a comprehensive overview of various NLP techniques and word embedding models, ranging from basic preprocessing steps to advanced machine learning algorithms. Using Sklearn and NLTK I learned how to clean and preprocess text data, represent words as dense vectors, and use these representations to train and evaluate different machine learning models. These techniques are essential for a wide range of NLP applications, including sentiment analysis, document classification, and language modeling.