Methodology and Model Evaluation Report

for Natural Language Processing Course
“Advanced Text Classification”

Introduction:

This report outlines the methodology followed in the development and evaluation of Machine learning with our project.

It includes data preprocessing, feature encoding, model selection.

Data Preprocessing:

The initial step involved loading and reading the data, followed by several preprocessing techniques to prepare the text for feature extraction:

- **Text Cleaning**: Removal of special characters to reduce noise in the data.

- **Tokenization:** Breaking down text into individual words or tokens.

- **Stop Words** Removal: Eliminating common words that provide little value in the context of text analysis.

- **Stemming**: Reducing words to their root form to standardize variations of the same word.

- **Label Encoding:** Converting categorical labels into numerical values to make them interpretable by machine learning algorithms.

Feature Encoding:

For the transformation of text data into a numerical format, two main techniques were employed

Word Embeddings:

- Utilizing pre-trained GloVe and Word2Vec models from the Gensim API to encode text into vector form.

- TF-IDF Vectorization: Applying Term Frequency-Inverse Document Frequency to emphasize important words which are more informative but less frequent.

Feature Scaling: The Impact of StandardScaler

When I used the StandardScaler to scale values, it improved the result in some cases like SVM in Glove word embeddings it got a 31% f1-score.

Model Selection and Evaluation

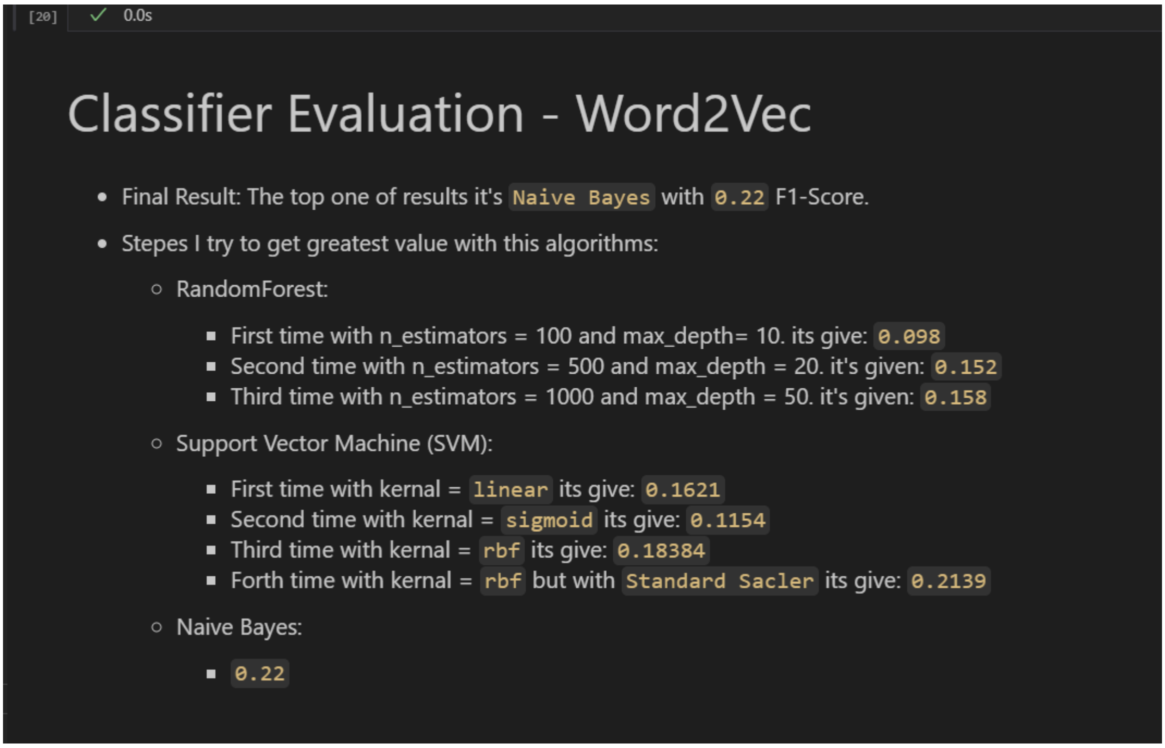
Models were selected based on their ability to handle high-dimensional sparse data and evaluated using the F1-score metric, which balances precision and recall:

* Naive Bayes
Achieved the best results with Word2Vec features, demonstrating an F1-score of 0.22.

* Random Forest:
Multiple configurations were tested. The best performance with GloVe was observed with n_estimators=500 and max_depth=20, achieving an F1-score of 0.172.

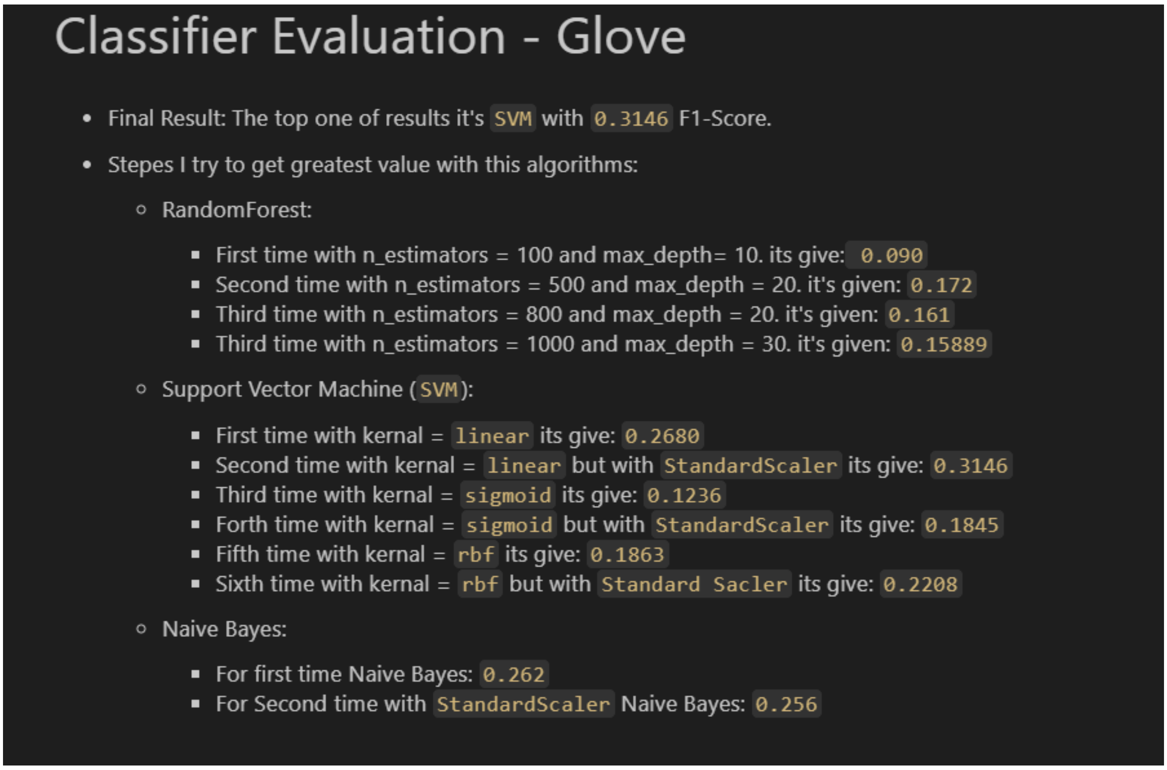
* Support Vector Machine (SVM)
The SVM showed significant variability based on the choice of kernel and scaling. The highest F1-score of 0.3146 was achieved using a linear kernel with standard scaling on GloVe features.

- This is a result from each algorithm in other models



Classifier Evaluation - Word2Vec

- Final Result: The top one of results it's `Naive Bayes` with `0.22` F1-Score.
- Stepes I try to get greatest value with this algorithms:
  - RandomForest:
    - First time with n_estimators = 100 and max_depth= 10. its give: `0.098`
    - Second time with n_estimators = 500 and max_depth = 20. it's given: `0.152`
    - Third time with n_estimators = 1000 and max_depth = 50. it's given: `0.158`
  - Support Vector Machine (SVM):
    - First time with kernal = `linear` its give: `0.1621`
    - Second time with kernal = `sigmoid` its give: `0.1154`
    - Third time with kernal = `rbf` its give: `0.18384`
    - Forth time with kernal = `rbf` but with `Standard Sacler` its give: `0.2139`
  - Naive Bayes:
    - `0.22`



Classifier Evaluation - Glove

- Final Result: The top one of results it's `SVM` with `0.3146` F1-Score.
- Stepes I try to get greatest value with this algorithms:
  - RandomForest:
    - First time with n_estimators = 100 and max_depth= 10. its give: ` 0.090`
    - Second time with n_estimators = 500 and max_depth = 20. it's given: `0.172`
    - Third time with n_estimators = 800 and max_depth = 20. it's given: `0.161`
    - Third time with n_estimators = 1000 and max_depth = 30. it's given: `0.15889`
  - Support Vector Machine (`SVM`):
    - First time with kernal = `linear` its give: `0.2680`
    - Second time with kernal = `linear` but with `StandardScaler` its give: `0.3146`
    - Third time with kernal = `sigmoid` its give: `0.1236`
    - Forth time with kernal = `sigmoid` but with `StandardScaler` its give: `0.1845`
    - Fifth time with kernal = `rbf` its give: `0.1863`
    - Sixth time with kernal = `rbf` but with `Standard Sacler` its give: `0.2208`
  - Naive Bayes:
    - For first time Naive Bayes: `0.262`
    - For Second time with `StandardScaler` Naive Bayes: `0.256`