PA1: Linear Models & Text Classification

Behrad Rezaie
COMP 550, McGill University, 261102374

***Abstract*—This project explores the classification performance of linear models (Logistic Regression, LinearSVC) in semantic and morphological differentiation.**

***Index Terms*—Linear Models, Sentiment, Morphology**

I. Introduction & Problem Setup

This experiment evaluates the application of Logistic Regression and LinearSVC for the task of differentiating text by semantic sentiment detection and morphology in plural noun detection.

The hypotheses are as follows: both models will perform very similarly, constrained by their linearity preventing complex understanding of text. For this reason, models should perform better than random guessing for sentiment detection, but not at high accuracy as complex relationships between words cannot be accurately determined. For plural detection, performance should be much higher, as certain morphological traits almost directly correlate with the presence of plurals. Both these tasks should be affected by different preprocessing methods, which in certain cases may potentially remove noise and increase performance, or remove too much information and lead to performance loss.

II. Dataset Generation

LLM tools were used to generate 200 sentences for each category of Negative/Positive Comments and Singular/Plural containing sentences without plurals. Each experiment's categories were combined, randomized, and split into training and test sets via a 90/10 split, with the train set also being used for validation during cross-validation.

III. Preprocessing & Feature Extraction

A number of preprocessing steps were experimented with, including the lowercasing of text, removal of punctuation, removal of common English stop words, stemming, and lemmatization. Results were recorded with and without the application of each of these methods, individually and collectively.

To extract features, the **TF-IDF-Vectorizer** tool from the scikit-learn library was used. This identifies and extracts relevant n-grams from a given corpus based on their computed TF-IDF score, discarding features with a DF score below 2 as configured.

For sentiment detection, the extractor was configured to extract word-level unigrams and bigrams. Additionally, a manually designed feature representing the count of "negative" words such as "bad, no, not, n't" in each sentence was also provided to the model.

For plural detection, the vectorizer was configured to work with character-level unigrams and bigrams, which extracted meaningful combinations of 1-2 character groupings within the corpus.

IV. Model Training & Parameter Exploration

Logistic Regression and LinearSVC models were trained on the dataset, with a grid search across their different regularization methods and strengths, through a 5-fold cross validation process.

V. Results and Conclusions

Across all experiments, both models performed similarly, indicating little effect in the choice of model on final performance.

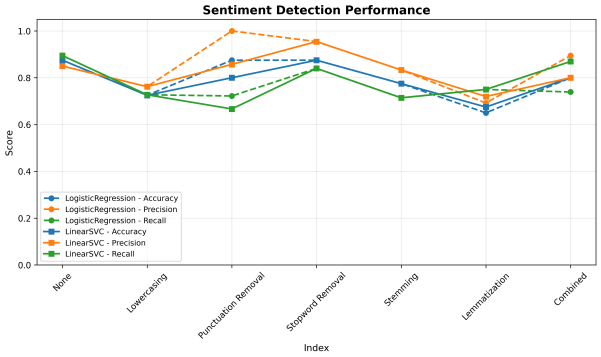

Fig. 1. Sentiment Classification Metrics vs Preprocessing Techniques

As seen in figures 1 and 2, performance varied significantly depending on the preprocessing steps applied, with little variance between the two types of models. Performance was measured with no preprocessing methods applied, one method applied in isolation at a time, and all methods applied simultaneously. Across the 7 experiments, sentiment detection accuracy mostly ranged between 75-80%, reaching a maximum of 87.5% when

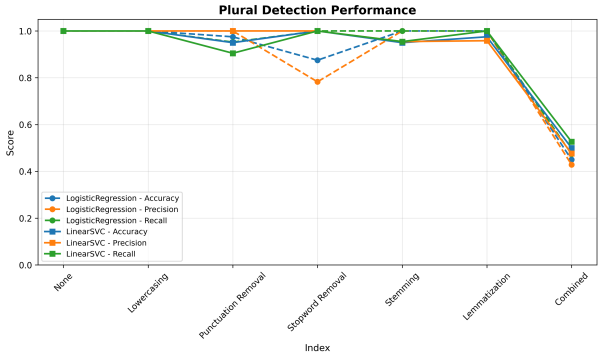

Fig. 2. Plural Detection Metrics vs Preprocessing Techniques

stopwords were removed, and a minimum of 65% with lemmatization applied. This indicates the model performed best when words without much semantic content were removed from the inputs, but suffered significantly when words were lemmatized into their canonical form, indicating these words lost too much information.

For plural detection, performance was at 90-100% for most experiments, dropping to 50% only when all preprocessing methods were applied simultaneously. This indicates the loss of too much information, likely due to the removal of key parts of words due to stemming and lemmatization, resulting in loss of information that would hint at the presence of plurals.

The weights of the features contributing the most to each classification were explored. For the model performing best on sentiment detection, terms such as *"unpleas","neg experi","poorli"* contributed the most to negative classification, whereas *"pleasant","exceed","posit experi"* contributed most to positive classifications. For plural detection, character n-grams such as *"re","s ","ese"* were the strongest indicators of plurals. This suggests the model learns plural forms of words as well as verb conjugations, such as *"are"* to indicate the subject being in plural form.

VI. Limitations of Experiment

This experiment relied entirely on LLM-generated data, which introduces bias in sentence structure and formation that may not be representative of real-world data. Additionally, this experiment is limited in scope to the exploration of only two linear models for relatively simple classification tasks. Performance on tasks requiring deeper language semantic understanding would be significantly more limited, requiring the use of more complex architectures. Finally, more powerful text processing techniques and features, such as part-of-speech tagging, were not explored. Their use could yield interesting results, enabling deeper understanding of text.