Enhancing Sentiment Analysis Using POS Tagging Report

Introduction

Sentiment analysis is a valuable application of natural language processing (NLP) that aims to determine the sentiment of a given text. In this report, I have explored the development of a sentiment analysis model by initially creating a POS tagger and improvising it. The primary focus is to compare the performance of a baseline model that does not utilize POS (Part-of-Speech) tags with an enhanced model that incorporates POS tag features.

Part 1: POS Tagger

The first part of my analysis involved the development of a POS tagger based on transition and emission probabilities. The Treebank corpus, available through NLTK, was used for my training and extracting these probabilities.
This process included:

- Counting the occurrences of tag transitions and tags in each sentence.
- Calculating transition and emission probabilities based on these counts.
- Storing the resulting probabilities in dictionaries (`transition_prob` and `emission_prob`).
- Developing the Viterbi algorithm from scratch and using it to predict the POS of a random sentence.

Part 2: Vanilla Sentiment Analyzer

In the second part, I implemented a Vanilla sentiment analysis model using the Movie Reviews dataset.
Steps I followed:

➔ Loading the dataset and structuring it as pairs of words and their corresponding labels (positive or negative).
➔ Splitting the dataset into training, validation, and test sets.
➔ Performing TF-IDF vectorization on the textual data.
➔ Training a Multinomial Naive Bayes classifier on the TF-IDF features and labels from the training set.
➔ Evaluating the classifier on both the validation and test sets

Part 3: Improved Sentiment Analyzer with POS Tags

The third part of my analysis is dedicated to enhancing the sentiment analysis model by incorporating POS tag features. Here, I outline the steps taken:

- Created a function to extract POS tags from tagged documents.
- Performed TF-IDF vectorization on the words in the dataset, similar to Part 2 task.
- Combined the TF-IDF word embeddings and standardized POS tag features using NumPy's `np.hstack`. This resulted in a combined feature matrix that includes both word-level and POS tag-level information.
- Preparing target variables for training, validation, and test sets.
- Trained a Multinomial Naive Bayes classifier on the combined feature matrix and labels from the training set.
- Evaluated the classifier on both the validation and test sets.

Comparison and Observations

Classification Reports:

POS-tag-enhanced model

```
test_predictions = classifier.predict(X_test_combined)
print("Test Accuracy:", accuracy_score(y_test, test_predictions))
print(classification_report(y_test, test_predictions))
```

```
Test Accuracy: 0.702
              precision    recall  f1-score   support

         neg       0.93      0.46      0.61       258
         pos       0.62      0.96      0.76       242

    accuracy                           0.70       500
   macro avg       0.78      0.71      0.69       500
weighted avg       0.78      0.70      0.68       500
```

Baseline Vanilla Sentiment Analyser:

```
test_predictions = classifier.predict(X_test)
print("Test Accuracy:", accuracy_score(y_test, test_predictions))
print(classification_report(y_test, test_predictions))
```

```
Test Accuracy: 0.732
              precision    recall  f1-score   support

         neg       0.89      0.55      0.68       258
         pos       0.66      0.93      0.77       242

    accuracy                           0.73       500
   macro avg       0.78      0.74      0.72       500
weighted avg       0.78      0.73      0.72       500
```

- **From the Reports we can see our POS enhanced model performs nearly equal to the baseline model** !.
- The reason for this might be the effectiveness of POS tags depends on their quality and relevance to the specific sentiment analysis task.
- The baseline model, which relies solely on TF-IDF word embeddings, provides decent sentiment analysis results.
- The enhanced model, incorporating POS tag features, offers an opportunity to capture syntactic information.

This report serves as a comprehensive overview of my work and provides insights into the development of sentiment analysis models with and without POS tag enhancements.

Thank you