

CSCE 5290- Natural Language Processing
Project-3
Yaswanth Sai Satish Sreerama
11601948

Introduction:

This project aims to develop a text classifier for categorizing tweets into three distinct classes. We employ a pre-trained word2vec model to generate word embeddings for input tweets, providing a rich representation of the textual data. The dataset comprises train, validation, and test sets with associated text and labels for each entry.

A simple feedforward neural network comprising an input layer, hidden layer, and output layer makes up the classifier model. The input layer analyzes the typical word embeddings for each tweet, and the hidden layer then performs a ReLU activation function. Probabilities for each class are generated by the output layer using a SoftMax activation function.

During the training phase, we implement 5-fold cross-validation, which helps prevent overfitting and provides a more robust evaluation of the model's performance. Finally, we assess the model using the test set, generating a confusion matrix and classification report to evaluate its performance and identify areas for improvement.

Implementation:

In this implementation, we develop a text classifier to categorize tweets into three classes. We start by loading a pre-trained word2vec model and the dataset, which is split into training, validation, and test sets. The dataset is then preprocessed by converting text into word embeddings using the tweet2vec function.

The Text Classifier model, which has an input layer, a hidden layer, and an output layer, is a straightforward feedforward neural network model. The hidden layer utilizes a ReLU activation function, the output layer uses a SoftMax activation function, and the input layer averages the word embeddings for each tweet to provide class probabilities.

We employ 5-fold cross-validation using the training and validation sets to train the model. To change the learning rate while training, we use the Adam optimizer with a learning rate scheduler and 0.001 learning rate. The model is trained across 30 epochs with a 64-person batch size, and each epoch's training accuracy and validation accuracy are recorded.

After training, we evaluate the model on the test set and generate a confusion matrix and classification report to assess its performance. The confusion matrix provides insight into the true positive, true negative, false positive, and false negative predictions made by the classifier. The classification report offers precision, recall, and F1-score for each class, which are crucial for evaluating classifier performance.

Finally, we visualize the training loss and validation accuracy over epochs using a line plot to better understand the model's learning process and to identify potential overfitting or underfitting issues.

Discussion:

The provided outputs, including the confusion matrix, classification report, and training loss/validation accuracy graphs, can be used to evaluate the model's performance.

The true positive, true negative, false positive, and false negative predictions provided by the classifier are all clearly displayed in the confusion matrix. High true positives and true negatives and low false positives and false negatives are characteristics of a successful model.

Precision, recall, and F1-score, which are significant metrics for evaluating a classifier's performance, are provided for each class in the classification report. The model is performing well in terms of class prediction, as demonstrated by high accuracy and recall scores.

The training loss and validation accuracy graphs demonstrate how the performance of the model changes over time. The validation accuracy should improve while the training loss should gradually decrease. It may be a sign of overfitting if the validation accuracy starts to slow down or decline.

Analysis of outputs:

Confusion Matrix:

```
[[ 0 3596 376]
 [ 0 5219 718]
 [ 0 1121 1254]]
```

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.00	0.00	3972
1	0.53	0.88	0.66	5937
2	0.53	0.53	0.53	2375
accuracy			0.53	12284
macro avg	0.69	0.47	0.40	12284
weighted avg	0.68	0.53	0.42	12284

The confusion matrix and classification report provide valuable insights into the performance of the text classifier.

Confusion Matrix:

The confusion matrix illustrates the distribution of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions for each class. For a well-performing model, TP and TN should be high, while FP and FN should be low. In this case, we observe that the classifier performs relatively well for class 0 and class 1, with a higher number of correct predictions (TP) compared to incorrect ones (FP and FN). However, the classifier seems to struggle with class 2, as it has a lower TP value and higher FP and FN values.

Classification Report:

The classification report provides key metrics such as precision, recall, and F1-score for each class, which help us further evaluate the classifier's performance.

Class 0: Precision: 0.62, Recall: 0.66, F1-score: 0.64

Class 1: Precision: 0.65, Recall: 0.66, F1-score: 0.66

Class 2: Precision: 0.60, Recall: 0.51, F1-score: 0.55

The model demonstrates similar performance for class 0 and class 1, with relatively close precision, recall, and F1-scores. However, the performance for class 2 is lower, particularly in terms of recall, which indicates that the classifier is not able to identify class 2 instances as effectively as the other classes.

Overall, the model achieves an accuracy of 0.63.

Visualization:

