[Applied Analytics Project Week 7 Assignment]

Major: Applied Analytics

Name: Heng Bai, Yuang Guo

1. The Model Approach Chosen and Its Suitability for the Prediction Task

For our project, XGBoost was selected as the model, applied with three different text representation methods: Count Vectorizer, TF-IDF Vectorizer, and Word2Vec embeddings. XGBoost is a gradient-boosted decision tree model that effectively handles complex classification tasks by combining the strengths of multiple trees to improve prediction accuracy. XGBoost's flexibility makes it an appropriate choice for text classification, where sequential learning can enhance model accuracy by focusing on instances misclassified in previous rounds. Each representation method captures different textual features: Count Vectorizer captures frequency, TF-IDF captures term importance, and Word2Vec encodes semantic meaning, helping us determine which representation best supports the task.

2. Complexity of the Modeling Approach

Our project's modeling approach involved significant complexity, particularly in optimizing XGBoost's parameters for three variations of text representations. XGBoost itself is computationally intensive due to its sequential tree-boosting process, and when combined with text data (often high-dimensional), the need for effective parameter tuning adds another layer of complexity. For each representation—Count Vectorizer, TF-IDF, and Word2Vec—separate models were trained and tuned, demanding substantial computational resources and model tracking. This complexity is necessary for thoroughly exploring the different feature representations and their effectiveness for the classification task.

3. Hyperparameters Evaluated and the Reasons for Their Selection

For our project, we evaluated three critical hyperparameters: learning rate, max depth, and number of estimators.

- **Learning Rate** was tuned to control how quickly or slowly the model adapts to patterns, helping to avoid overfitting by preventing overly aggressive learning.
- **Max Depth** of each tree was chosen to balance model flexibility with overfitting risks. Shallower trees (lower max depth) generalize better, while deeper trees can capture complex patterns but may overfit.
- **Number of Estimators** refers to the number of boosting rounds or trees in the ensemble. Setting this to a higher value allows the model to perform more corrections on prior errors but risks overfitting if not limited appropriately.

These hyperparameters were selected to balance predictive power and generalization, ensuring the model was adaptable without sacrificing accuracy or risking overfitting.

4. Identified Model Performance Metrics and Their Relevance

Our project evaluated each model's performance using accuracy, precision, recall, and F1-score across the training, validation, and test sets. Each metric provides unique insights:

- **Accuracy** gives a straightforward measure of the model's correctness across all classes.

- **Precision** and **Recall** assess class-specific performance, which is essential in multi-class classification to ensure balanced predictions.
- **F1-score** combines precision and recall, providing a more balanced metric that is useful for datasets with potential class imbalances.

These metrics were calculated for training, validation, and test sets to evaluate the model's ability to generalize beyond the training data. Given that this is a text classification task, these metrics were appropriate to measure both general accuracy and specific class performance.

5. Metrics Calculation on Training and Validation Datasets

For each model variation in our project, metrics were calculated on both training and validation datasets. This approach allows for an assessment of how well each model variation (Count Vectorizer, TF-IDF, Word2Vec) performs on unseen data while preventing overfitting. The code implemented these calculations by applying the model to the training and validation datasets and then using accuracy_score and classification_report to derive and output the results.

6. Calculated Metrics for All 3 Variations on Both Training and Validation Datasets

These metrics indicate that while each model was trained effectively on the data, Word2Vec embeddings yielded the best results on the validation set. By computing accuracy across both training and validation datasets, our project ensures that each model's generalization capacity is thoroughly evaluated, allowing for a more informed selection of the best-performing model.

7. Analysis of Training and Validation Metrics Changes Across Variations

Our project's analysis reveals distinct trends across the three feature extraction methods. The Word2Vec-based model demonstrated the highest validation accuracy at 72.57%, suggesting that embeddings were able to capture richer semantic information than the traditional count-based methods (Count Vectorizer and TF-IDF). The Count Vectorizer and TF-IDF models performed similarly on the validation set, with accuracies of 65.08% and 64.14%, highlighting that while they effectively capture word frequency and importance, they may lack the semantic depth that Word2Vec provides. This difference emphasizes the advantage of using dense embeddings in capturing context-dependent information, which is likely valuable for multi-class text classification.

8. Comparison of Performance Metrics for the Validation Dataset Across All 3 Variations

In our project, validation accuracy for the Word2Vec model reached **72.57%**, outperforming the Count Vectorizer (65.08%) and TF-IDF (64.14%) models. Word2Vec also achieved a more balanced F1-score, as evidenced in the individual classification reports. This comparison validates that Word2Vec offers a more robust solution for this multi-class classification task, likely because it preserves word context, enhancing the model's predictive quality in unseen data.

9. Table of 3 Variations Showing Training and Validation Metrics

Below is a table summarizing the training and validation metrics, providing a side-by-side comparison for all three variations.

Model Variation	Training Accuracy	Validation Accuracy	Validation F1-score
Count Vectorizer	High	65.08%	Based on class reports
TF-IDF	High	64.14%	Based on class reports
Word2Vec	High	72.57%	Based on class reports

10. Identification of the Best Model for the Week and Justification for Selection

The Word2Vec model was identified as the best-performing model in our project. Its superior validation accuracy and balanced F1-score metrics across classes make it a reliable choice, as it generalizes well and effectively captures nuanced textual relationships. Word2Vec's ability to encode word context likely contributed to its strong performance, especially in a task requiring nuanced text understanding. This model is recommended as the optimal choice for its demonstrated accuracy and generalization in handling multi-class text classification.