INTENSITY ANALYSIS

Given Objective

The objective of this project is to develop an intelligent system using NLP to predict the intensity in the text reviews. By analyzing various parameters and process data, the system will predict the intensity where its happiness, angriness or sadness. This predictive capability will enable to proactively optimize their processes, and improve overall customer satisfaction.

TABLE OF CONTENTS

1. Introduction

The Text Intensity Classifier is a machine learning pipeline designed to classify text into three intensity levels: angry, happy, and sad. This documentation provides a detailed overview of the design choices, performance evaluation, and future work considerations for the classifier.

2. Design Choices

2.1 Data Loading and Preprocessing:

- Data Sources: The classifier leverages three distinct datasets, each containing text samples associated with different emotional intensities: anger, happiness, and sadness. These datasets are loaded into memory using the Pandas library, facilitating easy manipulation and preprocessing.
- Labeling: To enable supervised learning, intensity labels ('angry', 'happy', 'sad') are assigned to each text sample in their respective datasets. This labeling step is crucial for training the classifier to recognize and differentiate between different emotional intensities.
- Data Concatenation: Once labeled, the individual datasets are concatenated into a single unified dataset. This consolidation step simplifies subsequent data processing tasks, such as feature extraction and model training, by providing a cohesive dataset encompassing all emotion categories.

2.2 Model Training:

- Classification Algorithms: The classifier employs three popular classification algorithms: Support Vector Machine (SVM), Random Forest, and Gradient Boosting. Each algorithm offers distinct advantages in terms of model complexity, robustness, and interpretability, thereby providing a diverse set of models for comparison.
- Feature Representation: Text data is inherently unstructured and requires transformation into numerical features for model training. To accomplish this, the classifier utilizes TF-IDF (Term Frequency-Inverse Document Frequency) vectorization. This technique converts text documents into sparse numerical vectors, where each feature represents the importance of a word in the document relative to the entire corpus.
- Model Training Pipeline: To streamline the training process and ensure consistency, a pipeline-based approach is adopted. This pipeline encapsulates both the TF-IDF vectorization and classification steps, facilitating seamless integration and reproducibility.

2.3 Evaluation and Visualization:

- Performance Metrics: The performance of each trained model is evaluated using multiple metrics, including accuracy, precision, recall, and F1-score. These metrics provide insights into the classifier's ability to correctly classify text samples across different emotion categories.
- Visualization: To aid in result interpretation and model comparison, the classifier generates visualizations such as confusion matrices and accuracy comparison plots. These visual representations offer intuitive insights into the distribution of predictions and the relative performance of different models.

3. Performance Evaluation

3.1 Model Accuracy:

- SVM: The SVM model achieves an accuracy of 79%. Accuracy measures the proportion of correctly classified samples out of the total number of samples in the test set.
- Random Forest: The Random Forest model achieves an accuracy of 78%. Random Forest is an ensemble learning technique that constructs multiple decision trees during training and outputs the mode of the classes as the prediction.
- Gradient Boosting: The Gradient Boosting model achieves an accuracy of 77%. Gradient Boosting sequentially trains weak learners (decision trees) and combines them to build a strong predictive model.

3.2 Classification Report:

Precision, Recall, F1-score, Support: The classification report provides a detailed breakdown of precision (the ratio of true positive predictions to the total predicted positives), recall (the ratio of true positive predictions to the total actual positives), F1-score (the harmonic mean of precision and recall), and support (the number of occurrences of each class in the test set) for each emotion category (angry, happy, sad) across all models.

3.3 Confusion Matrix:

True Positive, True Negative, False Positive, False Negative: Confusion matrices offer a comprehensive view of the classifier's performance by illustrating the number of true positive, true negative, false positive, and false negative predictions for each emotion category. These matrices enable the identification of specific areas of confusion or misclassification, aiding in model refinement and optimization.

4. Future Work

4.1 Model Improvement:

- Text Preprocessing: Further exploration of advanced text preprocessing techniques, such as stemming, lemmatization, or handling of special characters, could enhance the quality of feature representation and improve model performance.
- Algorithm Exploration: Experimentation with alternative machine learning algorithms and thorough hyperparameter tuning may lead to the discovery of more effective models with superior predictive capabilities.

4.2 Handling Imbalance:

Class Imbalance: Addressing potential class imbalance issues within the dataset through techniques such as oversampling, under sampling, or class weighting can mitigate biases and enhance the classifier's ability to generalize to all emotion categories.

4.3 Deployment:

- Production Deployment: Transitioning the trained model into a production environment for real-time predictions requires careful consideration of scalability, efficiency, and reliability. Implementation of robust deployment pipelines and monitoring mechanisms is essential to ensure seamless integration with existing systems.
- User Interface: Developing an intuitive user interface for interacting with the classifier can enhance usability and accessibility, enabling users to input text samples and receive emotion intensity predictions in a user-friendly manner.

4.4 Continuous Monitoring and Updating:

- Performance Monitoring: Establishing mechanisms for real-time performance monitoring enables proactive identification of model degradation or drift, facilitating timely intervention and model recalibration as needed.
- Data Updates: Regularly updating the classifier with new data ensures its relevance and adaptability to evolving linguistic patterns and emotional expressions. Continuous training on fresh data enhances the classifier's predictive accuracy and robustness over time.

5. Conclusion

The Text Intensity Classifier represents a versatile tool for analyzing and categorizing text based on emotional intensity. Through careful design choices, rigorous evaluation, and considerations for future enhancements, the classifier demonstrates promising performance and scalability for real-world applications. By embracing ongoing advancements in machine learning and natural language processing, the classifier remains poised to evolve and adapt to emerging challenges and opportunities in text analysis and sentiment classification.