Model Optimization And Tuning Phase Report

Date	15 November 2024
TeamID	739909
Project Name	Unlocking the Minds: Analyzing Mental Health with NLP
Maximum Marks	10 Marks

Model Optimization and Tuning Phase:

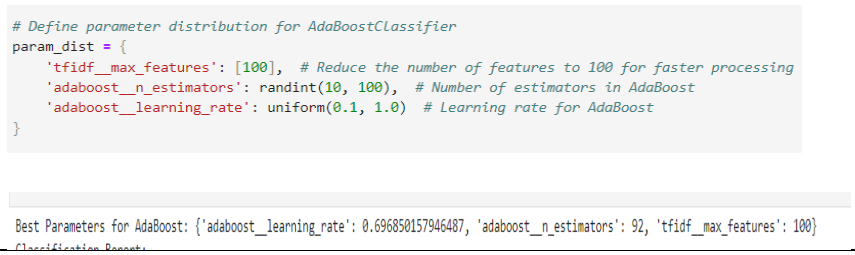
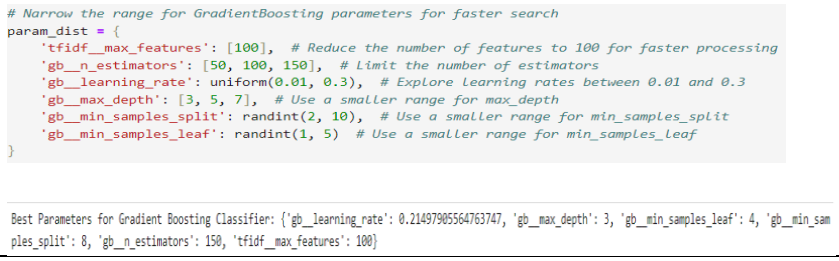
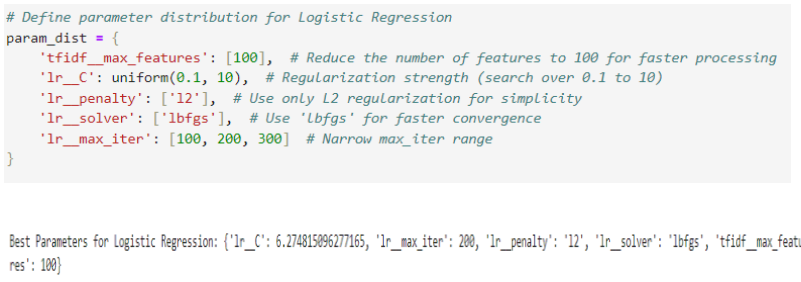
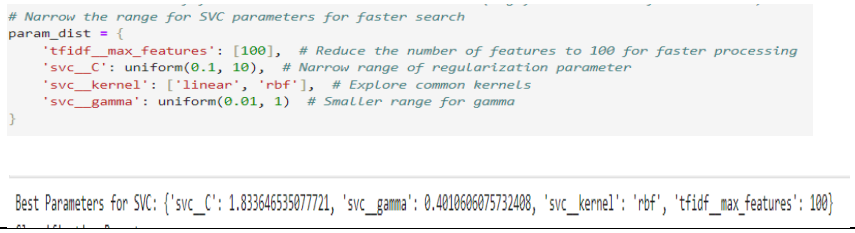
The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (8 Marks):

Model	Tuned Hyperparameters

Decision Tree Classifier	`# Narrow the range for DecisionTree parameters for faster search`
	`param_dist = {`
	`'tfidf__max_features': [100], # Reduce the number of features to 100 for faster processing`
	`'dt__criterion': ['gini'], # Fixed to 'gini' for faster exploration`
	`'dt__max_depth': [5, 10, 15], # Narrow the max_depth range`
	`'dt__min_samples_split': randint(2, 10), # Use a smaller range for min_samples_split`
	`'dt__min_samples_leaf': randint(1, 5) # Use a smaller range for min_samples_leaf`
	`}`
	
	Best Parameters for Decision Tree: {'dt__criterion': 'gini', 'dt__max_depth': 10, 'dt__min_samples_leaf': 3, 'dt__min_samples_split': 8, 'tfidf__max_features': 100}

Random Forest Classifier	`# Parameter distribution for RandomForestClassifier`
	`param_dist = {`
	`'tfidf__max_features': [100], # Reduce the number of features to 100 for faster processing`
	`'rf__n_estimators': [50, 100, 200], # Number of trees in the forest`
	`'rf__max_depth': [5, 10, 15], # Maximum depth of the tree`
	`'rf__min_samples_split': randint(2, 10), # Minimum samples required to split an internal node`
	`'rf__min_samples_leaf': randint(1, 5), # Minimum samples required to be at a leaf node`
	`'rf__bootstrap': [True, False] # Whether bootstrap samples are used when building trees`
	`}`
	
	Best Parameters for Random Forest: {'rf__bootstrap': False, 'rf__max_depth': 15, 'rf__min_samples_leaf': 4, 'rf__min_samples_split': 6, 'rf__n_estimators': 200, 'tfidf__max_features': 100}

AdaBoost Classifier	
Gradient Boosting Classifier	
Logistic Regression	
Support Vector Classifier	

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Support Vector Classifier (SVC)	The Support Vector Classifier (SVC) was selected for its effectiveness in handling high-dimensional textual data, making it particularly suitable for mental health analysis. Its ability to classify complex patterns using kernel-based transformations ensures high accuracy in identifying nuanced text features. Additionally, SVC's robustness to overfitting provides reliable predictions even with limited labeled data, aligning well with the project's objectives in "Unlocking the Minds: Analyzing Mental Health with NLP."