USMAN INSTITUTE OF TECHNOLOGY

Affiliated with NED University of Engineering & Technology, Karachi

Department of Computer Science

B.S. Computer Science

Project Proposal

Batch-2021

Predicting Acceptance Rates for Programming Problems

By

Shaheer Muhammad Shahbaz	21B-060-CS
Osama Khursheed	21B-159-CS
Shaheer Adil	21B-095-CS
Areeba Amir	21B-133-CS

Objective

This project aims to predict the acceptance rates of programming problems using a dataset containing metadata such as difficulty level, related topics, company associations, and user engagement metrics. The task will involve comparing the processing times required for data preprocessing and model training on a single processor versus a parallel/distributed computing setup.

Scope of Work

1. **Data Preprocessing**
    - Cleaning and standardizing the dataset by handling missing values and encoding categorical variables.
    - Feature engineering to generate derived metrics such as likes-to-dislikes ratios and topic frequency.
    - Scaling numerical features to improve model performance.
2. **Predictive Modeling**
    - Train a regression model to predict acceptance rates based on selected features.
    - Evaluate the model's performance using metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).
3. **Parallel or Distributed Processing**
    - Use parallel frameworks (e.g., Dask or PySpark) to optimize data preprocessing and feature engineering.
    - Distribute model training across multiple cores or nodes using TensorFlow Distributed or joblib with Scikit-learn.
4. **Performance Evaluation**
    - Compare the time taken for preprocessing and training tasks on a single processor versus a parallel/distributed setup.
    - Compute the speedup achieved with parallel processing.

Dataset

The LeetCode dataset provides comprehensive metadata for programming problems, including:

- **Features**: Attributes such as problem difficulty, number of likes and dislikes, related topics, associated companies, and detailed descriptions.
- **Target Variable**: The acceptance rate, which represents the percentage of successful submissions for each problem.

The dataset was sourced from Kaggle and can be accessed at the following link:
[LeetCode Problem Dataset]

Performance Metrics and Expected Outcomes

1. **Processing Time**: A significant reduction in time for data preprocessing and model training.
2. **Speedup**: A quantifiable improvement in performance using parallel or distributed processing.
3. **Model Accuracy**: A regression model with acceptable prediction accuracy, evaluated using MAE and RMSE.

Benefits

- **Time Efficiency**: Faster analysis of large datasets, enabling real-time insights for competitive programming platforms.
- **Scalability**: A distributed approach to handle future expansions of the dataset.
- **Data Insights**: Useful predictions for LeetCode users, administrators, and educators to identify problem difficulty trends.

Technologies to Use

1. **Languages**: Python.
2. **Libraries**: pandas, NumPy (for single processor), Dask, PySpark (for parallel processing).
3. **Machine Learning**: Scikit-learn, XGBoost, or LightGBM for regression.
4. **Visualization**: Matplotlib and Seaborn for presenting insights.