**MGT 683 MSBA Capstone Status Report**

| | |
|---|---|
| **Name:** | **Harika Yenuga** |
| **Report Timeframe:** | **November 11, 2024 - November 17, 2024** |
| **Project Title:** | **Onboarding Experience Optimization & Skill-Based Internal Job Matching for Employee Mobility** |

**Progress summary:**

***Work Done:***

1. **Model Development**:
   - **Random Forest Classifier**: Achieved an accuracy of 20%. Evaluation shows room for improvement, especially in balancing precision and recall across classes.
   - **Decision Tree Classifier**: Marginal improvement with an accuracy of 22%. F1-scores indicate moderate performance in specific classes.
   - **Support Vector Machine (SVM)**: Accuracy of 17%, limited by imbalanced classes and poor precision in most labels.
   - **CatBoost Classifier**: Accuracy of 22%, with better performance in certain classes compared to Random Forest.
2. **Clustering**:
   - Implemented **KMeans** with k=3 to group employees by skills. Cluster labels generated successfully, providing insights into skill gaps.
3. **EDA**:
   - Analyzed missing values, summary statistics, and feature distributions. Key insights revealed uneven data representation and class imbalance.

***Work in Progress:***

Fine-tuning models for better classification outcomes:

- Hyperparameter tuning for Random Forest and CatBoost using Grid Search.
- Feature scaling and balancing class distributions.

Refining clustering models:

- Evaluating cluster quality using silhouette scores and cohesion metrics.
- Beginning development of a recommendation engine using collaborative and content-based filtering methods.

Job-Role Matching:

- Extracting features like 'Skills', 'PerformanceRating', and 'JobSatisfactionScore'.
- Preparing to implement a recommendation system (content-based/collaborative).

Model Evaluation:

- Evaluating models using accuracy, precision, recall, and RMSE/MAE.

**Next steps:**

In the coming week 5, I will focus on:

Model Training & Feature Engineering:

- Feature Engineering: I'll keep extracting relevant features, such as Skills, Performance Rating, and Job Satisfaction, to improve both job-role matching and skill-based clustering. These features are crucial because they reflect the key attributes that influence job-role suitability and employee skills.

- Data Splitting: I'll divide the dataset into training and testing sets to ensure proper model evaluation. This is essential because the training set allows the model to learn, while the testing set helps me evaluate its performance on unseen data.

- Model Training: I will train Random Forest, Decision Trees, and K-Means Clustering models on the training data. Random Forest and Decision Trees are used for classification, while K-Means will group employees based on their skills.

**Hyperparameter Tuning:**

I'll apply Grid Search to tune the Random Forest and CatBoost models. This process involves testing different combinations of hyperparameters (like the number of trees, maximum depth, etc.) to find the best settings for higher accuracy and performance.

**Model Evaluation & Interpretation:**

**Performance Metrics:** I'll assess the classification models using standard metrics such as accuracy, precision, recall, and F1-scores. These will help me understand the effectiveness of each model and how well it identifies employee skill gaps and job suitability.
**Clustering Assessment:** I'll use silhouette scores to evaluate the quality of the clusters generated by K-Means. These scores help assess how well-separated the clusters are, ensuring that the skill-based groupings are meaningful.

**Problems encountered/Bottlenecks:**

Data Normalization Issues:

- I faced difficulties normalizing the JobSatisfactionScore and PerformanceRating variables. These scores required specific scaling to work well with the models, but the normalization process didn't always give expected results. As a result, model performance was affected, and this highlighted how sensitive these scores can be to different preprocessing techniques.

Data Integration:

- Merging multiple datasets from Kaggle was more time-consuming than expected. The datasets had to be adjusted to align columns properly with the problem I am solving. This process was challenging but necessary to ensure the data was structured correctly for the models.

Time Management:

- To improve efficiency, I reduced the dataset size to 1,100 rows. Although this limits the data available for training, it helps keep the project manageable by speeding up the training process and ensuring more efficient use of resources.

**Reflection:**

For this past three weeks, During my research, I came across several studies focusing on **onboarding optimization** and **job-role matching** using machine learning. Most studies emphasize the importance of **feature selection** and **data preprocessing** for improving model accuracy. They suggest using models like **Random Forest** and **SVM**, along with techniques like **hyperparameter tuning** and **ensemble methods**, to improve predictions for employee job matches.

However, there is a noticeable gap in integrating **employee skill data** with **internal mobility systems** in real-time, which is where my project stands out. Existing literature lacks detailed models that combine **skill-based matching** and **personalized job recommendations**. Additionally, many studies fail to address the challenge of **class imbalance** and how it impacts model performance in job-role predictions, something I have encountered in my work.

In summary, while the literature provides valuable insights, there's still a need for more research on **real-time, scalable models** that can enhance internal employee mobility by integrating various factors like **skills**, **performance**, and **job satisfaction**.