Kaggle DSML Survey Analysis & Job Recommendation Web App

1. Dataset Overview

This project uses three datasets from Kaggle's Annual Machine Learning and Data Science Surveys (2020-2022), encompassing responses from diverse professionals in the data science and machine learning industry. These datasets include a range of variables from demographics to professional experience and learning preferences, offering a detailed view of the field.

2. Preprocessing for job recommendation

This section outlines the development of the preprocessing pipeline, which systematically transforms raw data into an optimized format for the job recommendation model.

2.1 Preprocessing Steps

The preprocessing phase was meticulously architected to ensure the integrity and utility of the data for our recommendation system. The following key steps were instituted:

I. **Integration of Survey Queries**: The consolidation process involved merging columns from each dataset that corresponded to identical survey questions, ensuring coherence and uniformity across all data points. Then merged data collected over three years into a unified dataset.

II. **Categorical Variable Processing**: In this critical step, the categorical variables were meticulously cleaned to streamline the dataset for machine learning algorithms. This involved stripping extraneous characters like parentheses, segmenting compound entries at commas to isolate distinct items, and converting numerical ranges into single median values.

III. **Missing Value Handling:** Rigorous scrutiny was applied to identify and address missing values within the dataset. Entries lacking crucial information that could not be accurately reconstructed were excluded to maintain the integrity of the analysis.

2.2 Encoding and Normalization

The next stage involved encoding categorical variables and normalizing numerical features:

I. **MultiLabel Binarization**: Multiple-answer categorical variables were converted into binary matrices using MultiLabel Binarizer, preparing them for machine learning algorithm interpretation.

II. **Normalization**: Numerical features were normalized to a 0-1 range using MinMaxScaler, ensuring balanced feature contribution during model training and preventing skewing by any single feature.

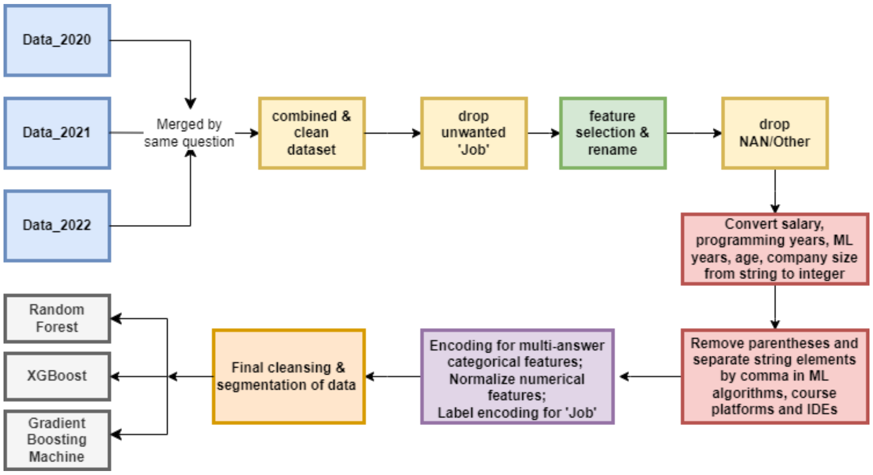

Data_2020
Data_2021
Data_2022
Merged by same question
combined & clean dataset
drop unwanted 'Job'
feature selection & rename
drop NAN/Other
Convert salary, programming years, ML years, age, company size from string to integer
Remove parentheses and separate string elements by comma in ML algorithms, course platforms and IDEs
Encoding for multi-answer categorical features; Normalize numerical features; Label encoding for 'Job'
Final cleansing & segmentation of data
Random Forest
XGBoost
Gradient Boosting Machine

Figure 1 Data Processing Pipeline

3. Exploratory Data Analysis (EDA) insights:

This section will conduct an in-depth Exploratory Data Analysis (EDA) of the data science domain, uncovering trends and correlations to provide insights into the current dynamics of the DSML industry.

3.1 Languages used on a regular basis

The bar chart (Fig.2) shows Python's dominance at 30% and SQL's solid presence at 17% in the programming market, both growing steadily, emphasizing their roles in DSML. Other languages like R, C, and Java hold a steady 6-9% share. Bash and MATLAB encounter slight declines, possibly due to open-source alternatives, while Julia maintains a small but consistent 1% share, indicating stable language preferences in tech over time.

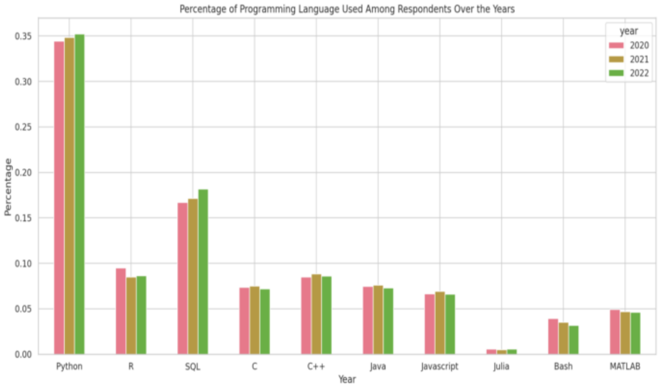


Figure 2 Programming Language Used Over the Years

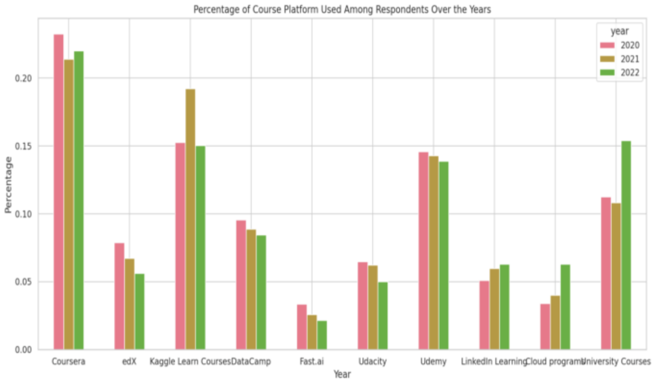


Figure 3 Course platform used over the years

3.2 Course Platform chosen

Fig.3's bar chart shows Coursera leading at 20%, with Kaggle and Udemy trailing at 15%, probably favored for their wide course ranges. A 5% jump in university course use in 2022 points to increased value placed on formal education in DSML, while continuous declines in other platforms suggest a potential consolidation trend around top provider.

3.3 Company size

Pie charts (Fig.4) exploited to depict a clear trend, where the proportion of respondents from very small companies (0-49 employees) is shrinking, while there is an increase in representation from larger companies, especially those with over 10,000 employees. This shift suggests a possible industry consolidation or a movement of the workforce towards larger enterprises within the data science and machine learning fields.

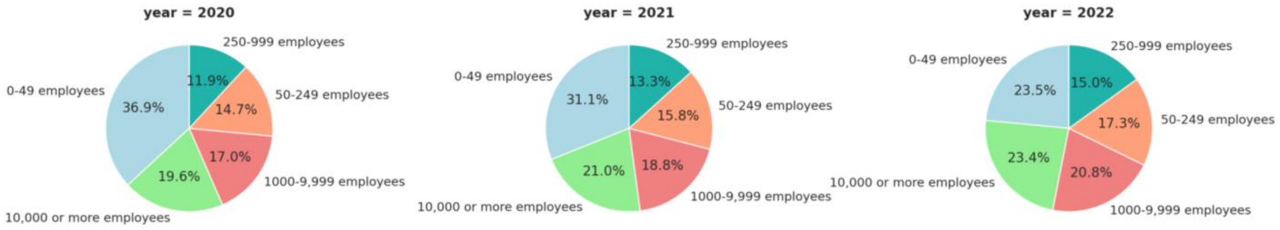


Figure 4 Company size over years

3.4 Years of programming

Evaluating the programming experience (Fig. 5) of survey participants, it's evident that individuals with less than three years of experience make up the majority in the DSML career sector. Moreover, their proportion is on the rise, reflecting the growing opportunities available in the data science arena.

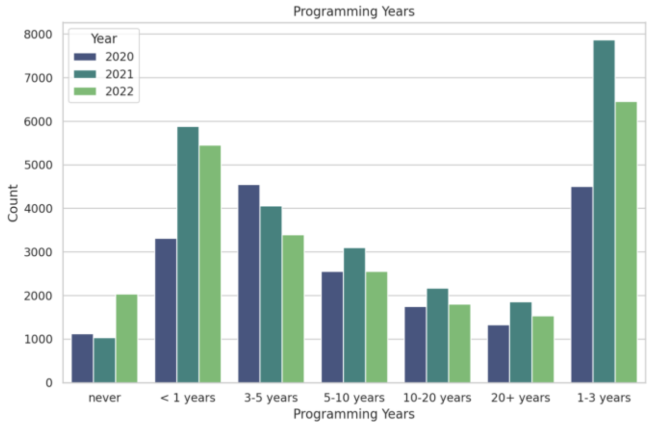


Figure 5 Programming years summary

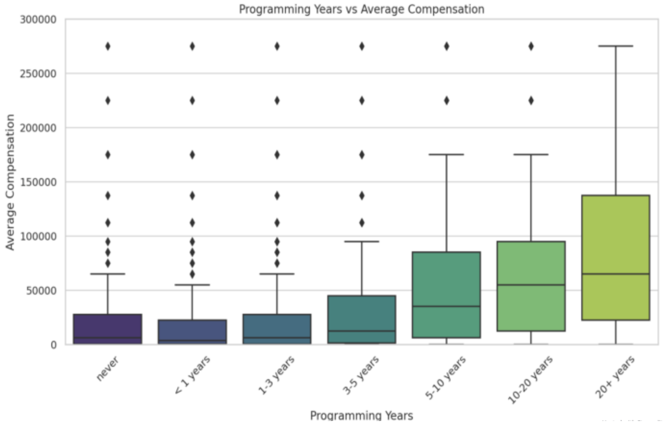


Figure 6 Programming year vs. Yearly Compensation

3.5 Salary vs. Years of programming

The boxplot above (Fig. 6) suggests a positive correlation between programming experience and salary, with earnings tending to rise as experience grows. However, variability in higher experience brackets and the occurrence of outliers indicate that other factors beyond years of programming may also significantly impact compensation like job role, industry, or locations.

3.6 Company Size vs. Average compensation

Heatmap (Fig. 7) was used here finds that larger companies generally pay higher salaries, with a clear gradient showing more top earners at companies with over 10,000 employees. However, there's substantial salary variability across all company sizes, and mid-sized companies also offer competitive high salaries.

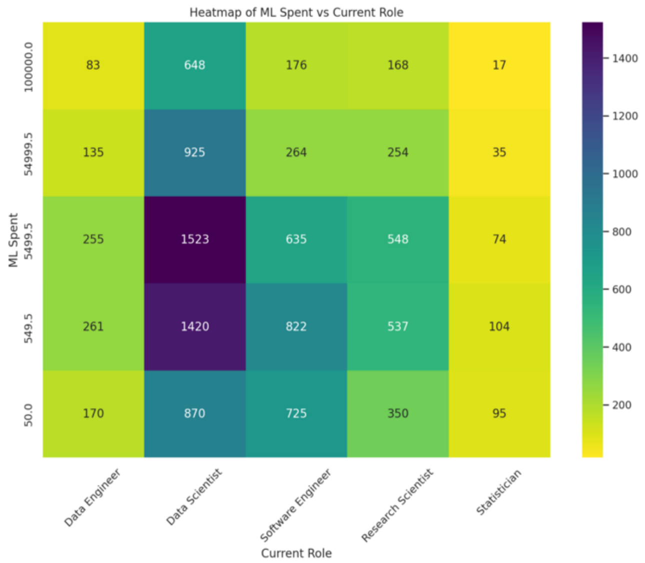


Figure 7 Company size vs. Yearly Compensation

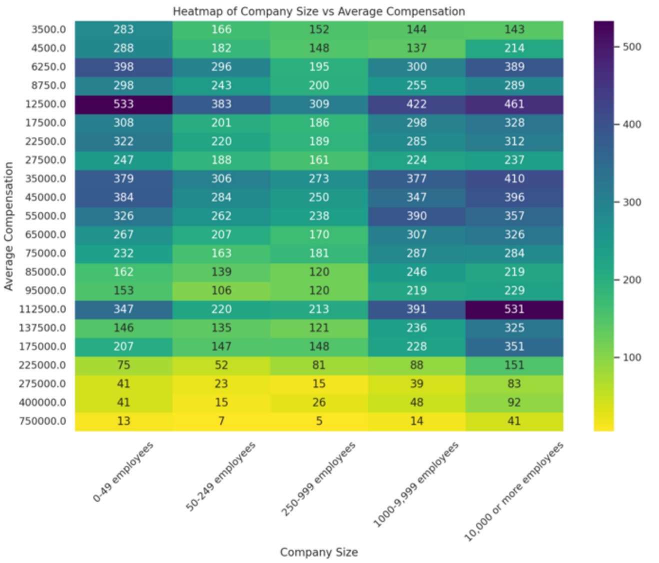


Figure 8 ML spent vs. Current role

3.7 ML spent vs. Current role

The heatmap (Fig. 8) illustrates that Data Scientists are the principal spenders on ML, with substantial costs incurred mostly by this group. In contrast, other roles such as Data and Software Engineers show more modest spending. High investment in ML education is not widespread, suggesting that significant spending on learning ML is more specialized, likely within roles that are heavily ML-focused.

4. Job Recommendation Tasks

The background progress in designing the job recommendation app would be elaborated in this section.

4.1 Feature Selection

To optimize the job recommendation system, the SelectKBest method is employed to isolate the most crucial features from the dataset. This strategic selection facilitates a leaner model by concentrating on the most predictive attributes. It also informs the design of the app's user interface by highlighting the qualifications and experiences that are most impactful in predicting job compatibility.

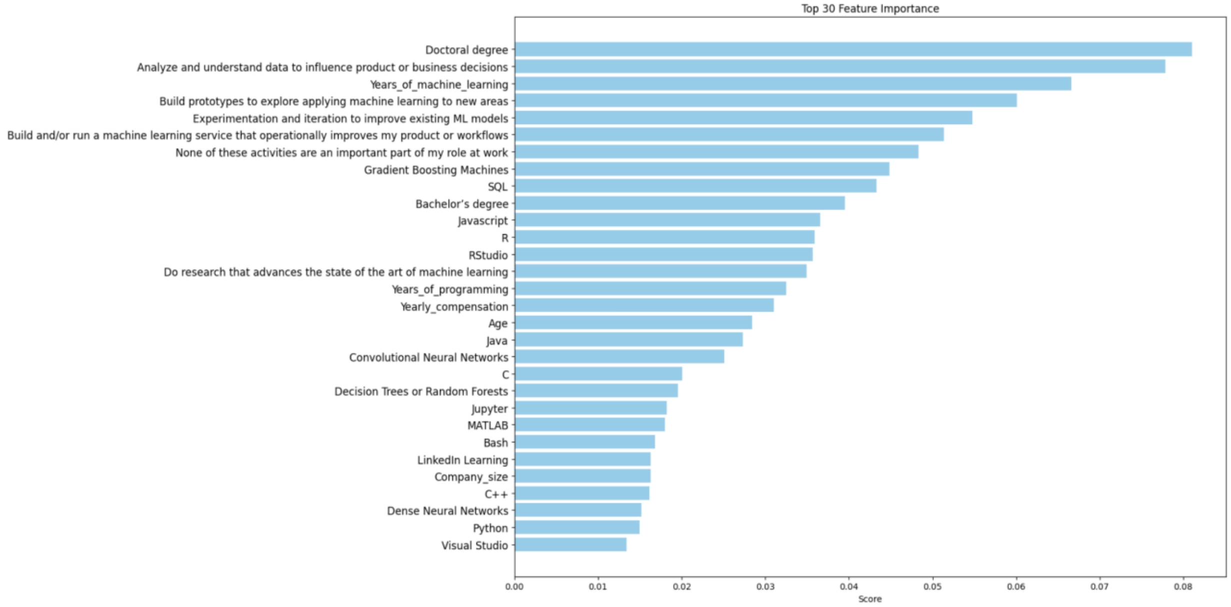
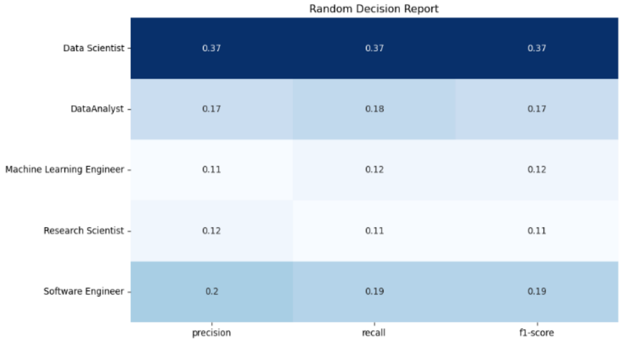


Figure 9 Top 30 Features Selected

The graph (Fig. 9) reveals those academic qualifications such as a 'Doctoral Degree', practical experience like 'Years of machine learning', and analytical abilities exemplified by 'Analyze and understand data to influence product or business decisions' carry significant weight in the feature importance ranking, following up with mastered programming languages.

4.2 Baseline Model - Random Decision

The comparative analysis was initiated by establishing a baseline model. A random decision approach was employed to offer a basic benchmark, against which the performance improvements from more sophisticated algorithms could be evaluated. Ideally, Figure 10 illustrates the accuracy of the baseline model in comparison to random choice. Which overall accuracy is approximately 23% with equal class distribution.



	precision	recall	f1-score
Data Scientist	0.37	0.37	0.37
DataAnalyst	0.17	0.18	0.17
Machine Learning Engineer	0.11	0.12	0.12
Research Scientist	0.12	0.11	0.11
Software Engineer	0.2	0.19	0.19

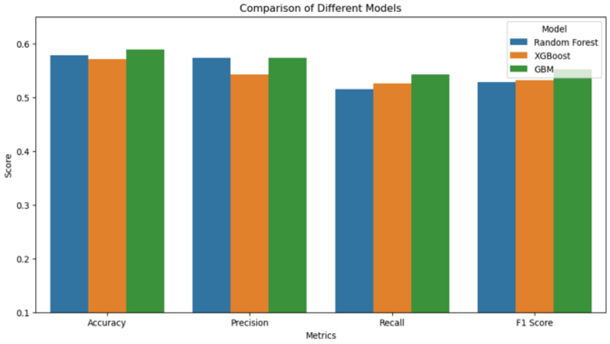


Figure 10 Random Choice Model Performance *Figure 11 Model Comparison Metrics*

4.3 Model Comparisons

The investigation into the predictive capabilities of different algorithms included an extensive comparison of several high-performing models in classification tasks. The RandomForest, XGBoost, and Gradient Boosting Machine (GBM) models were chosen evaluated first, compared against each other as well as the baseline, to gauge their predictive effectiveness. A bar chart (Fig. 11) was plotted here to demonstrate the variance in accuracy, precision, recall, and F1 scores among the evaluated models. The GBM model stands out with the highest of all metrics. It could be observed that all models have accuracy higher than the random decision, showcasing the algorithms are effectively learning from the data, rather than making arbitrary predictions, which is a positive sign of model validity and potential utility in practical applications.

4.4 Predicted Job Selection

Initially, titles such as 'student' and 'currently not employed' were excluded from consideration for job recommendations. This left 16 distinct job titles (Fig 12), from which those with fewer than 500 occurrences were further eliminated, resulting in nine remaining job categories. The GBM model, noted for its superior performance as illustrated before, showed lower-than-anticipated accuracy for 'Statistician', 'Data Engineer', 'Business Analyst', and 'Product Manager' (Fig.13), which also had a smaller representation in the dataset, leading to a low overall accuracy of 48%. Consequently, these titles were removed, focusing on the five most populous job categories, which correspondingly improved the model's accuracy, and the result would be elaborated in the next section.

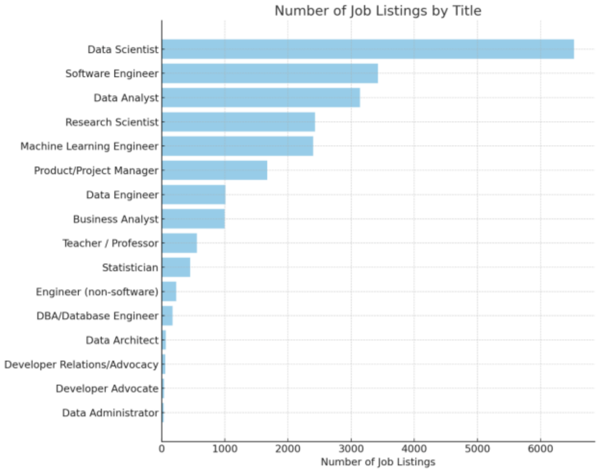


Figure 12 Number of Jobs by Title

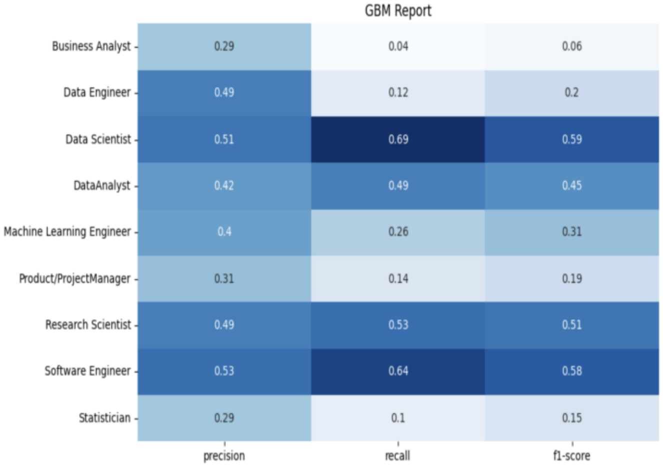


Figure 13 9 Jobs GBM Model Result

4.5 Final result

The final optimized GBM model's performance is highlighted in Figure 14, followed with a confusion matrix result (Fig.15), showcasing enhanced predictive accuracy for the top five job categories, which increased to an overall accuracy of 59% and is 10% improved than before. This refined approach in summary, backed by data-driven feature selection and model comparison, ensures that the job recommendation app can effectively match candidates with suitable positions.

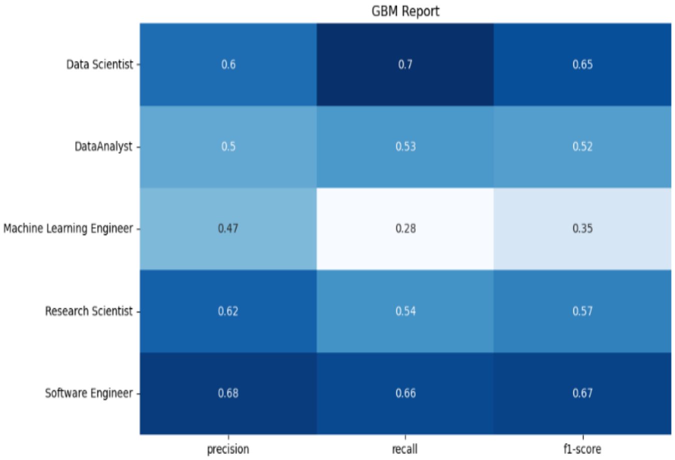


Figure 14 GBM performance metrics

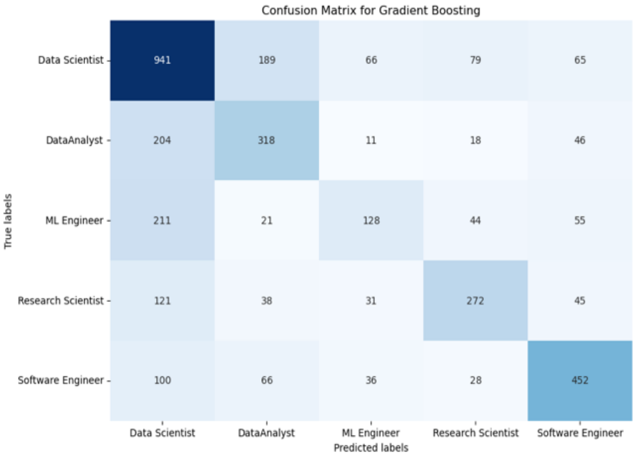


Figure 15 Confusion Matrix for GBM

5. Reflection and Conclusion

5.1 Insights from Data to Decision

The journey from raw data to a functional recommendation system has been both challenging and enlightening. Reflecting on the process, the importance of meticulous data preprocessing and the careful selection and tuning of machine learning models became clear.

5.2 Challenges Encountered

One of the more significant challenges was managing the diverse and sometimes sparse dataset, which necessitated thoughtful feature selection and unnecessary rows drop-out. Additionally, the process of translating categorical data into a machine-understandable format highlighted the need for creative encoding methods.

5.3 Learning Experiences

The project provided us with valuable practical experience in handling real-world data and understanding the subtleties involved in modeling complex relationships. We learned the significance of not only model accuracy but also the other performance metrics like F1, ensuring that the recommendations provided by the system are justifiable and transparent.

5.4 Future Directions

Looking forward, we aim to enhance our system by incorporating more dynamic features and perhaps exploring the use of deep learning techniques like neural networks, which might capture nuances in the data that traditional machine learning models may overlook. We also recognize the potential for expanding our model to include real-time labor market data, further refining the accuracy and relevance of the job recommendations.

5.5 Conclusive Thoughts

In conclusion, the development of the Job Role Recommendation App represents a convergence of data science and practical application, with the potential to significantly impact users' professional trajectories. While our model currently provides a strong foundation, the pathway for continuous improvement and adaptation remains an exciting prospect for future endeavors.

6. Reference:

[1] 2020 Kaggle Machine Learning &amp; Data Science Survey. [Online]. Available: https://www.kaggle.com/c/kaggle-survey-2020/.

[2] 2021 Kaggle Machine Learning &amp; Data Science Survey. [Online]. Available: https://www.kaggle.com/c/kaggle-survey-2021/.

[3] 2022 Kaggle Machine Learning &amp; Data Science Survey. [Online]. Available: https://www.kaggle.com/c/kaggle-survey-2022/.

[4] “Streamlit docs,” Streamlit documentation. [Online]. Available: https://docs.streamlit.io/.

[5] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.