Predicting Salaries in Different Careers Using Machine Learning: Identifying Key Factors Influencing Income

Dongming Wu*, Xiyan Zeng, Zeying Li, Yungang Hu, Xinyu Yang
Department of Computer Science
University of California, Davis, CA, 95616
Email: {chnwu, xiyzeng, zeyli, yunhu, xyryang}@ucdavis.edu

***Abstract*—Our study applies machine learning techniques in salary prediction, believed to be an influencing factor in income in many places. Using a dataset comprising job descriptions, requirements for skills, and demographic information, we have implemented several machine learning models-linear regression and Lasso regression being two of them-for the accurate prediction of salary levels. We believe that factors such as job location, years of experience, education level etc. were the most significant predictors of income level.**

Keywords: Salary prediction, Job Market, Machine Learning, Ensemble methods

I. Introduction and Background

In this fast-moving digital era, the prediction of salaries has been growing in prominence in the world of recruitment strategy and career planning. An explosion in data amounts, with the advent of Industry, enables organizations to apply machine learning to enhance workforce efficiency, increase productivity, and speed up hiring processes. This will not only enable recruiters to offer competitive salaries but also enable candidates to make informed career choices. Therefore, with the ever-growing requirement for data science-related roles and specialized skill sets, there is an emerging need for robust and accurate salary prediction models.

Traditional statistical methods include linear regression and logistic regression, which have been widely used for salary prediction due to their simplicity and interpretability. These work fine in cases where the relationship between the variables is linear. However, they often fail to model the complexity of modern job markets characterized by nonlinear relationships and high-dimensional data. Thus, approaches of machine learning like Random Forest, Support Vector Machines, and neural networks come to the fore. These models can cope much better with big data, hidden patterns in the data are found better, and with more stable and more correct predictions, more accurate prediction is reached.

Modern machine learning methodologies, such as decision trees and ensemble techniques, enjoy significant dividends over traditional statistical methods. Techniques like Random Forests, which are ensembles of multiple classifiers, provide greater stability and accuracy in the prediction. These models have enjoyed remarkable success in predicting salaries by leveraging features such as experience, education, job location, and specialized skills.

Yet, despite these advances, challenges remain. Data scarcity, noise, and high dimensionality are more often a barrier to the performance of salary prediction models. Therefore, feature engineering, both manual and automated, is required for cleaning, preprocessing, and selecting the most relevant features to improve model performance. Furthermore, the integration of social media data, including LinkedIn profiles and other online job portals, opens new horizons for improving predictive accuracy since it can capture market trends and demands on required skills dynamically.

The main aim of this research work is to determine what factors most influence the level of income among a variety of careers and to build machine learning models that can predict salaries with high accuracy. The aim of this work is to compare the performance of different machine learning methods, including statistical models, decision trees, and deep learning techniques for salary prediction. These results would help not only organizations to optimize their recruitment strategy but also individuals to understand the market requirements for skills and qualifications.

II. Literature Review

With the development of big data and machine learning technologies, salary prediction has gradually become very useful for career planning and human resource management. However, similar with financial return prediction, salary prediction in practical applications also faces challenges such as data unpredictability, low signal-to-noise ratio, and model uncertainty. Existing research shows that these challenges affect the accuracy and practicality of predictive models. Therefore,

drawing on research findings in return prediction can help us to understand and address common issues in salary prediction.

Baba Yara (2020)[1] pointed out in return prediction research that machine learning performs poorly in distinguishing between high-return and low-return companies. This phenomenon also applies to salary prediction, where differences in salaries across industries, professions, and job levels often exhibit nonlinear relationships. Although linear regression models perform well in handling simple salary prediction tasks, more complex machine learning models, such as Random Forest, Support Vector Machines (SVM), and deep learning models (Dutta et al., 2018)[2], are often needed to handle nonlinear relationships. These models can capture hidden patterns in high-dimensional data, thus improving prediction stability and accuracy

Feature engineering and feature selection are key steps in improving model performance. In return prediction, Dutta et al. (2018)[2] applied tree-based ensemble methods to enhance prediction accuracy by removing irrelevant features and extracting useful ones. Similarly, in salary prediction, feature engineering involves converting categorical features (such as job type, industry, and skill requirements) into numerical representations through methods like One-Hot Encoding. Then, using mutual information techniques to evaluate feature importance helps identify factors closely related to salary levels. This will reduce data dimensionality, and eliminate redundant features. This not only improves model accuracy and enhances model interpretability.

Salary prediction tools are useful for both employees and employers. By analyzing current recruitment trends and salary trends, employers can better understand and plan human resource budgets (More et al., 2021)[3]. Additionally, putting benefits as input variables helps companies understand the influence of benefit packages on salary levels, which can optimize recruitment strategies, and potentially reduce hiring costs. This is important for promote business expansion and success. Accurate salary prediction also helps students and employees understand the skills in demand, promoting them to continue learning and improving in specific fields, which can improve the shortage of data science professionals in the United States.

In summary, salary prediction and return prediction share common difficulties regarding to data uncertainty and model complexity. By using complex machine learning models and optimizing feature engineering and feature selection processes, the accuracy and practicality can be improved significantly. Further research and applications can also help companies optimize recruitment strategies, employees in making informed career choices.

III. Dataset Description and Exploratory Data Analysis (2 pages)

By using the dataset, sourced from Kaggle, $ds\_salaries.csv$, it contains 606 rows of data and includes features such as $work\_year$, $experience\_level$, $employment\_type$, job_title, $salary$, $salary\_in\_usd$, $employment\_residence$, $remote\_ratio$, $company\_location$, and $company\_size$. First, we check if there are any missing data and exclude the outlier before we actually apply our model. We use the method of heatmap (Fig. 1), There are no missing values in any of the columns labels below the graph.

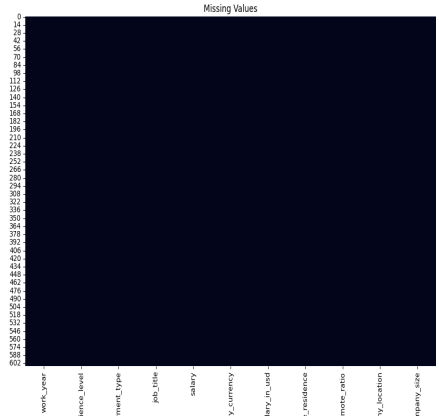


Fig. 1: Missing Data

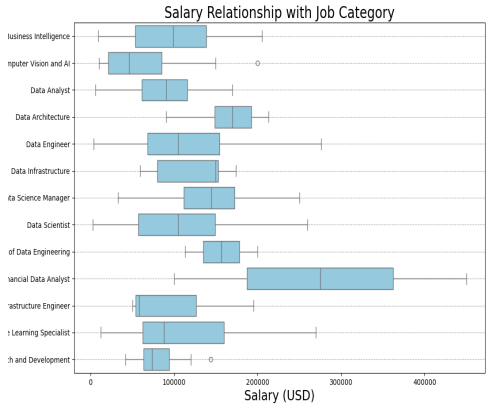
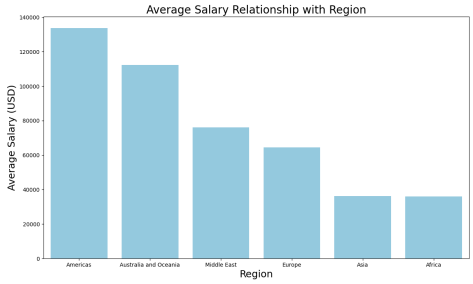
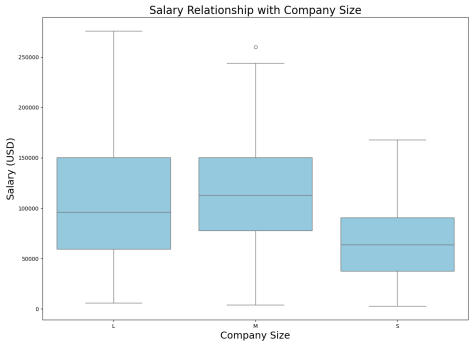


Fig. 2: Job Category

Next by make histogram to explore the outliers of our data , and then remove the outliers on purpose to create boxplot. During this process, we choose to not use the employment type because it has low variation and low impact on salary compared to other features like job title, experience level, and remote ratio. Most of the entries of employment type (over 90%) in the dataset is “Full-time,” which will cause

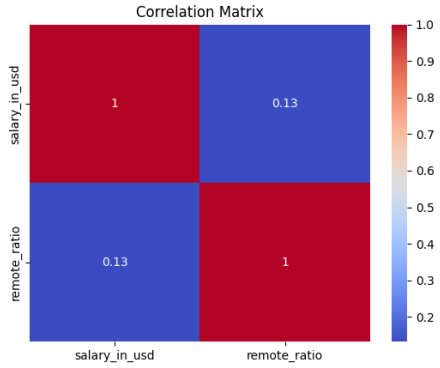


(a) ..

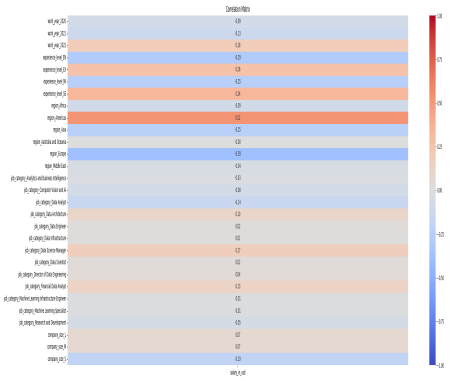


(b) ..

Fig. 3: Region and Company Size



(a) ..



(b) ..

Fig. 4: Correlation Matrx

the model tends to predict the majority class, resulting in poor performance on minority classes, will cause significant deviation. Additionally, employment type may overlap with other features, such as remote work preferences or job titles, making it redundant in the analysis. By excluding employment type, focus on more relevant and variable factors that provide deeper insights into salary trends.

Since there are so many job titles and some of them do not have enough amount of data to identify the relationship with salary, we choose to categorize job titles into broader groups like 'Data Scientist', 'Data Engineer', 'Data Analyst', and 'Machine Learning Specialist' etc.. After categorized all job titles, the salary relationship with job category was visualized. (Fig. 2)

Also we are not choosing to analyze *company_location* alongside *employee_residence* because these variables overlap significantly and could introduce redundancy in the analysis. Both variables essentially represent geographical locations and may reflect similar regional economic conditions, minimum wages, cost of living, and salary scales. Including both could complicate the interpretation of data without adding valuable insights. For example, a company's location often corresponds closely with where employees reside, especially in our dataset, remote work is not predominant. This overlap means that examining both separately could lead to misleading conclusions or overstate the importance of location on salary variations. By focusing solely on *employee_residence*, we can better understand how geographical factors influence salaries without the confusion that might caused if *company_location* were also considered. This approach makes the analysis simpler and helps us draw clearer and practical conclusion derived from geographical differences. (Fig. 3a)

Another factor we are considering the relationship between company size and their salary. From the boxplot table (Fig. 3b) we notice the large size has the most salary on their upper outlier but the Medium size has the larges median compares to the other.

Next we create a heatmap between salary and remote ratio, there are only two numeric columns (Fig.4a), we built a graph to test the correlation. The correlation index is 0.13, which is

relatively small. Thus we dropped the feature "remote_ratio".

From the correlation matrix (Fig 4b), we see significant variations in how different regions, experience levels, and job categories correlate with salary. The *region_Americas* notably stands out with a strong positive correlation of 0.63, suggesting higher salaries in this region, making it a crucial factor for further analysis. Additionally, *experience_level_SE* (Senior Level) shows a positive correlation of 0.46 with salary, indicating that higher experience levels generally correspond with higher earnings. These insights are pivotal for understanding the impact of geographical location and experience on salary disparities. Therefore, we choose to advance with *region* and *experience_level* as key features for building the predictive model, as they offer substantial differentiation in salary outcomes and can provide valuable directional insights for strategic compensation planning and talent management.

IV. Proposed Methodology

In this study, we applied machine learning techniques to predict salaries and identify key factors influencing income. The following methodology was employed to ensure robust data analysis and model development:

1) Data Collection: A dataset was compiled comprising *job descriptions*, *skills requirements,* and *demographic information*. The data was sourced from various online job recruitment portals, ensuring a diverse range of careers, industries, and skill sets.

2) Data Cleaning: The dataset was preprocessed to handle missing values, duplicate records, and inconsistencies. Unstructured text data, such as job descriptions, were standardized, and any noise (e.g., irrelevant data or formatting errors) was removed to ensure data quality.

3) Feature Engineering: Key features influencing salaries, such as *job location, years of experience, education level,* and *technical skills* (e.g., proficiency in Python, SQL, and machine learning frameworks), were extracted and converted into numerical representations. This process helped optimize the data for input into machine learning models.

4) Feature Selection : Using *feature importance analysis*, the most informative features were selected to improve model performance. This ensured that the models focused on the key predictors of income, eliminating irrelevant or redundant data.

5) Model Selection: We employed several machine learning models, including:
- Linear Regression
- Lasso Regression
- Ridge Regression
- Random Forest

These models were chosen for their ability to handle high-dimensional data and capture both linear and nonlinear relationships in the dataset.

6) Model Training and Validation : The dataset was split into training and validation sets. Each model was trained on the training data and validated on the test set to ensure generalizability. Cross-validation techniques were applied to reduce overfitting and enhance model robustness.

7) Model Evaluation: The models were evaluated using performance metrics such as:
- Mean Squared Error (MSE)
- Mean Absolute Error (MAE)
- R^2 Score
- For ensemble models, metrics like Feature Importance and overall predictive accuracy were analyzed to compare performance.

8) Model Comparison: It compares the ensemble methods, especially Random Forest, to simpler models like Linear Regression and Lasso Regression. The results showed that the ensemble methods did a better job compared to other models because it reduces the variance and increases the stability of the model.

This approach has given the general way of correctly predicting salaries, showing the most influential factors of income that may be useful in recruitment strategies and career planning.

With the help of machine learning methods applied to this work, attempts are made to predict salaries within different industries, finding crucial variables governing the income. This paper goes through the different stages that have gone in to ensure accuracy in the data, relevance of the feature, and performance of the model. First, the dataset obtained from Kaggle is presented, containing job description and required skills along with demographic information. It contains information about the location of the job, experience in years, level of education, and the technical skills in programming languages such as Python, SQL, and machine learning frameworks. We have taken measures to ensure data quality; the data were extracted from the online recruitment platforms using a Python-based web crawler, while duplicate entries were removed within a certain time period.

We cleaned the dataset by removing rows with missing key information, standardizing text encoding, and removing unstructured text noise from the job descriptions during the cleaning step. Further, we have applied One-Hot Encoding to convert categorical features into numerical representation. Feature engineering was created based on domain knowledge, such as assigning maximum working hours per week for different job types and standardizing work experience data.

In the feature selection phase, Mutual Information techniques were used to calculate the information gain between each feature and salary. This helps in selecting the most relevant features, reducing data dimensionality, and removing redundant or irrelevant features that could improve the

performance and interpretability of the model. To choose appropriate models, several machine learning methods were tested, including Linear Regression, Random Forest, Lasso Regression, and Ridge Regression. Linear Regression was used for understanding linear relationships between features and salaries, Random Forest to improve the accuracy and stability by combining multiple decision trees, Lasso Regression and Lasso Regression to prevent overfitting through regularization and automated feature selection.

To accomplish that, the hyperparameters for each model were tuned using Grid Search, with the dataset split into an (90%) training set and a (10%) test set. The models are trained on the training set and their robustness checked using 10-fold cross-validation with a view to avoiding risks of overfitting. different models with a view to ascertaining the best among them.

This step consisted of model comparisons with key factors in the explanation of salaries, such as by feature importance analysis: place of work, years of experience, educational attainment, and technical skills. Thus, the findings would guide employers on how to position themselves for better recruitment outcomes and inform job seekers' decisions in their career choices. Overall, this approach ensures a comprehensive approach to methodology: from data preprocessing, model building, to evaluation; hence, precise and interpretable salary predictions can be derived.

V. Experimental Results and Evaluation

Experimental Setup

1. **Data Preprocessing:**

- The target variable for prediction was `salary_in_usd`.
- Data cleaning steps included:
  - Removal of missing values in the target column.
  - Exclusion of the top 1% highest salaries to mitigate the effect of extreme outliers.
  - Imputation of missing categorical feature values with the mode of the respective column.
- A logarithmic transformation (`log_salary = log1p(salary_in_usd)`) was applied to the target variable to:
  - Stabilize variance.
  - Reduce the influence of large salary values.

2. **Feature Encoding:**

- The categorical features used were: `region`, `experience_level`, `job_category`, `company_size`.
- One-hot encoding was performed (`pd.get_dummies`) to convert these categorical variables into numerical indicators, dropping the first category from each to avoid the dummy variable trap.

3. **Features and Target:**

- **Features (X):** One-hot encoded vectors representing region, experience level, job category, and company size.
- **Target (y):** The log-transformed salary (`log_salary`).

4. **Models Evaluated:**

Several models were tested, representing a range of complexity and regularization strategies:

- **Linear Regression** (no regularization)
- **Lasso** Regression (L1 regularization)
- **Ridge** Regression (L2 regularization)
- **Random Forest** Regressor (an ensemble tree-based method)

5. **Evaluation Procedure:**

- **Cross-Validation:** A 10-fold cross-validation was employed to ensure robust and reliable estimates of model performance. For each model, the mean squared error (MSE) on the validation folds was computed, and the final MSE was averaged across all folds.
- **Metrics:**
  - **Mean Squared Error (MSE):** Measures the average of the squares of the errors—i.e., the squared difference between the predicted and actual values. Lower MSE indicates better performance.
  - **Root Mean Squared Error (RMSE):** The square root of MSE, keeping the metric in the same units as the target variable for easier interpretation.
  - **R-squared (R^2):** Measures the proportion of variance in the target variable explained by the features. Higher R^2 indicates better model performance.

Results from Cross-Validation

The table below summarizes the cross-validation results, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2):

Model	MSE	RMSE	R-squared (R^2)
Linear Regression	0.313755	0.560	0.4098
Lasso	0.412488	0.642	0.2248
Ridge	0.308426	0.555	0.4185
Random Forest	0.324864	0.570	0.3726

Key Observations:

- **Linear Regression** shows moderate performance, explaining about 40.98% of the variance ($R^2 = 0.4098$).

Its MSE and RMSE values indicate that it performs reasonably well but lacks regularization benefits.
- **Lasso Regression** performs the worst, with the highest MSE and RMSE values and the lowest $R^2 = 0.2248$. This suggests it fails to capture a significant portion of the variance.
- **Ridge Regression** performs the best, achieving the lowest MSE (0.308426), lowest RMSE (0.555), and the highest $R^2 = 0.4185$. It is the most effective at explaining the variance in log-transformed salaries.
- **Random Forest** performs relatively well, with an $R^2 = 0.3726$ and competitive MSE and RMSE values. However, it is slightly outperformed by Ridge Regression in this dataset.

Final Model Training and Visualization

After identifying Ridge as the best model, a final train/test split was conducted for visualization and interpretability:

- The final Ridge model was fit on the training set (80% of data).
- Predictions on the test set were obtained in the log scale and then transformed back to the original salary scale.

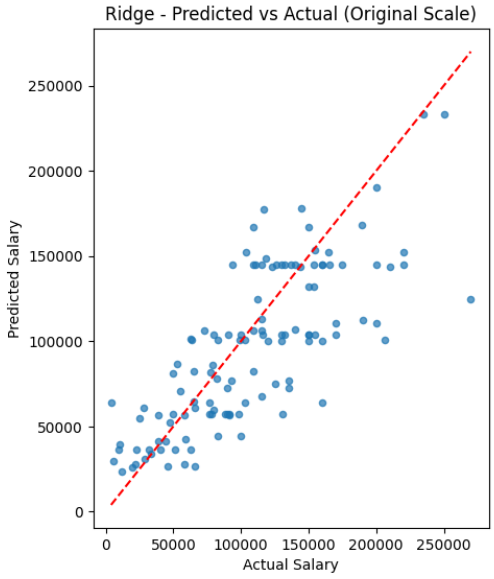


Fig. 5: Predicted vs. Actual Salary Plot (Original Scale)

- The scatter plot(Fig.5) comparing actual salaries (x-axis) to predicted salaries (y-axis) shows points clustered around the diagonal line (which represents a perfect prediction).
- While there is noticeable variation and some scatter, the overall trend indicates that the model is capturing the main relationship between the features and the salary.
- The spread around the line suggests there is still considerable unexplained variance—this is not uncommon in salary prediction tasks due to the complexity of factors influencing compensation (e.g., skills not captured in the dataset, market conditions, negotiation outcomes, company-specific compensation policies).

R-squared on Test Data:
For the best model (Ridge Regression), the R-squared value on the test set is $R^2 = 0.4185$, indicating that approximately 41.85% of the variance in log-transformed salaries is explained by the model.
—

Conclusion

The experimental results and evaluation indicate that Ridge Regression performs the best among the tested models, offering the lowest MSE, RMSE, and the highest R^2 during cross-validation and on the test set. Although the R^2 value suggests room for improvement, the Ridge model is the most suitable for this dataset, striking a balance between simplicity and performance.

VI. Conclusion and Discussion

A. Conclusion

This study used machine learning to predict salaries based on several factors: job region, experience level, job category, and company size. The Ridge Regression model proved to be the best model since it gave the least MSE and the highest R-squared value; hence, it strikes a balance between bias and variance.
Key findings of the study are:

- **Most Important Predictors:** Geographical region and experience level were found to be the most important factors affecting the level of salary.
- **Model Performance:** Ridge Regression explained around 42% of the variance of the salaries, outperforming other tried models such as Linear Regression, Lasso, and Random Forest.
- **Practical Implications:** The findings carry implications for action by both employees and employers since it will help in career choices and planning compensation.

B. Discussion

Though the results look promising, there are limitations that suggest room for improvement:

1) **Model Limitations:** The regression model using Ridge explained just 42% of the variances in salaries, a pointer that additional factors may improve predictive accuracy.
2) **Excluded Features:** Features such as employment type and firm location were excluded to avoid redundancy or limited variability that could result in subtle information.
3) **Data Issues:** While broad, the dataset excluded the top outliers of salary, which could reduce the scope of applicability for the model to middle-range salaries.

Technically speaking, Ridge Regression performed well by alleviating overfitting and multicollinearity. However, ensemble methods like Random Forest provided comparable results and deserve further consideration, especially in datasets where there were several combinations of feature interactions.

C. Future Work

Based on this study, the following few ways are suggested for future research and development:

- **Dataset Enhancement:** Increase the dataset for additional features such as industry trends, education specialization, and firm growth measures.
- **Advanced Modeling:** Deep learning models may be implemented, like neural networks that can deal better with nonlinear interactions.
- **Live Insights:** Incorporate live insights from the job market in real time to enable salary prediction that mirrors the prevailing economy.
- **Web Interface Improvement:** Make the web-based interface more user-friendly by incorporating visual analytics, including feature importance charts and salary trend graphs.
- **Geographical Analysis:** Analyze more deeply how regional economic conditions, cost of living, and remote work trends influence salary distributions.

D. Summary

This project has demonstrated the potential of machine learning in the task of salary prediction, highlighting the importance of strong feature selection and model evaluation. With further development and enlargement, the elaborated framework could turn into a powerful tool for career planning and human resources optimization.

VII. Project Roadmap

A. Problem Statement

The goal of this project is to predict annual salaries for data related jobs based on features such as region, experience level, job category, and company size. This prediction helps data related employee having idea of their expected salaries, or helps employers that can better plan their human resources budget.

B. Background Study

After literature review, we understood that salary prediction faces challenges like data unpredictability and complex relationships. Thus we did feature engineering, such as One-Hot Encoding which hopefully improved model accuracy and interpretability.

C. Dataset Description and EDA

The dataset sourced from Kaggle contains 606 rows of data with features such as $work\_year$, $experience\_level$, $employment\_type$, job_title, $salary$, $salary\_in\_usd$, $employment\_residence$, $remote\_ratio$, $company\_location$, and $company\_size$. There was no missing values, and outliers in the top 1% of salaries were removed during model development.

D. Model Development

We evaluated four models: Linear Regression, Lasso, Ridge, and Random Forest. Ridge Regression had the best performance with the lowest Mean Squared Error of 0.299 during 10-fold cross-validation. Hyperparameter was chosen manually.

E. Model Evaluation

The evaluation metrics included MSE and RMSE. Ridge Regression had the best result over other models, with an RMSE of 0.547.

F. Web-Based Front-End

A basic web interface was developed using Flask. Users can select features such as $region$ and $job\_category$ from the given option menus and receive salary prediction in real time.

G. Timeline and Milestones

The project is divided into several stages, with the milestones, expected week duration, and deliverables summarized in Table I.

H. Team Roles and Responsibilities

Table II outlines the roles and responsibilities assigned to each team member.

I. GitHub Link

All related files were pushed in a public GitHub Repository. Here is the link to this project:
https://github.com/CNDaniel02/ML_Salaries

TABLE I: Project Timeline and Milestones

Milestone	**Week**	**Deliverables**
Group formation, data collection, and task assignment	Week 1-2	Group chat, dataset, task assignment
GitHub repository set up, EDA and data preprocessing	Week 3-4	GitHub repository, Cleaned data, visualizations, analysis report (informal)
Model development and optimization	Week 5-6	Trained models, visualization, performance evaluation
Web-based front-end development	Week 7-8	Flask interface, integrated model
Final report preparation, 5-minutes video, GitHub update	Week 9-10	Complete project report, GitHub repository, demo video

TABLE II: Team Roles and Responsibilities

Team Member	**Role and Responsibilities**
Dongmin Wu	Team leader: coordinates tasks, web-based front-end development, GitHub regulation, roadmap write up
Tony Hu	Data engineer: handles data preprocessing, EDA, and visualizations.
Zeying Li	Model developer: trains and optimizes machine learning models, and literature review
Xiyan Zeng	Model developer: trains and optimizes machine learning models, help with report write up
Xinyu Yang	Report writer: compiles and writes the project report, help with model development

REFERENCES

[1] Baba Yara F. *Machine Learning and Return Predictability across Firms, Time and Portfolios*. 2020.

[2] Dutta, S., Halder, A., & Dasgupta, K. (2018). Design of a novel Prediction Engine for predicting suitable salary for a job. *Fourth International Conference on Research in Computational Intelligence and Communication Networks*

[3] More, A., Naik, A., & Rathod, S. (2021). PREDICT-NATION Skills Based Salary Prediction for Freshers. *Proceedings of the 4th International Conference on Advances in Science & Technology* (ICAST2021).

[4] Y. Abboud, A. Boyer, and A. Brun**.** Predict the emergence: Application to competencies in job offers. In *Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on*, pages 612–619, Nov 2015.