Exploring the Impact of Unlabeled Data on Job Salary Prediction: A Comparative Study of Machine Learning Techniques

Anonymous

1 Introduction

Predicting job salaries accurately is crucial for job seekers and employers alike. Various machine learning techniques have been employed for this purpose, with different levels of success. However, the potential of using unlabeled data in conjunction with labeled data for salary prediction may remain under-explored. This study aims to assess the impact of unlabeled data on the performance of several machine learning algorithms, including K-Nearest Neighbors (KNN), Logistic Regression (LR), Support Vector Machines (SVM), and Random Forests (RF).

We compare the strengths and weaknesses of these algorithms when applied to a dataset containing both labeled and unlabeled job salary information [1]. By evaluating their performance, we seek to answer whether effectively combining labeled and unlabeled training data can improve job salary prediction. The remainder of this paper is organized as follows: Section 2 provides a literature review on job salary prediction and machine learning techniques; Section 3 describes the methodology; Section 4 presents the results; Section 5 discusses the findings; and Section 6 concludes the paper and outlines future research directions.

2 Literature Review

In the context of predicting job salaries, various machine learning techniques have been explored to achieve accurate predictions. Lothe et al. proposed a salary prediction system using a linear regression algorithm with second-order polynomial transformation to predict employee salaries based on relevant features. They found that the model achieved an accuracy of 76% with a mean squared error (MSE) of 357, meeting their goal of reducing the MSE to below 360. The approach considered factors such as job type, degree, major, industry, years of experience, and miles from the metropolis to make accurate predictions. The results of the system were calculated using suitable algorithms and compared with other algorithms in terms of standard scores and curves, such as classification accuracy, F1 score, ROC curve, and precision-recall curve. The researchers aimed to further enhance their model by adding a graphical user interface and saving and reusing the trained model in future work [3]. This study serves as a foundation for exploring the use of various machine learning algorithms, including supervised, unsupervised, and semi-supervised learning, in predicting job salaries and analyzing the effectiveness of these algorithms in different contexts.

In another study, Sananda Dutta, Airiddha Halder, and Kousik Dasgupta (2018) focused on predicting salaries for job advertisements without specified salaries. They aimed to help fresh graduates predict potential salaries for different companies in various locations. The cornerstone of this study was a dataset provided by ADZUNA. The model they developed was capable of predicting precise values, indicating its potential for further application in salary prediction tasks [2].

3 Method

3.1 Supervised Machine Learning

3.1.1 Logistic Regression

Logistic Regression (LR) is a statistical technique used to analyze datasets with multiple outcome categories. While it is typically employed for binary classification problems, logistic regression can also be adapted to address multi-class classification tasks using approaches such as the one-vs-rest (OvR) strategy. In the case of our job salary prediction problem, which involves 10 distinct salary classes, logistic regression can be applied utilizing the one-vs-rest method. This process entails training individual binary logistic regression models for each class, designating one class as the positive class and the remaining classes as negative. For a given input, the class with the highest probability is chosen as the final prediction. The capacity of logistic regression to manage high-dimensional data and adapt to multi-class classification challenges renders it an appropriate choice for this task.

3.1.2 Support Vector Machines

Support Vector Machines (SVM) are a class of algorithms that aim to find the optimal separating hyperplane between different classes. SVMs are well-suited for this problem, as they can handle high-dimensional data and are effective in dealing with multi-class classification tasks. The flexibility of SVMs to work with various kernel functions allows them to capture complex relationships between features and salary classes.

3.1.3 Random Forests

Random Forests (RF) are an ensemble learning method that constructs multiple decision trees during training and combines their predictions to improve overall performance. This method is particularly useful for this problem due to its ability to handle high-dimensional data and handle multi-class classification tasks effectively. Additionally, Random Forests are robust to overfitting and can provide insights into feature importance, which can be valuable for understanding the most significant factors in job salary prediction.

3.2 Semi-supervised Machine Learning

We also explore the use of semi-supervised learning techniques to leverage the available unlabeled data for job salary prediction. Specifically, we employ a strategy that resembles self-training, which is a form of semi-supervised learning that utilizes a pre-trained model's predictions on the unlabeled data.

In our approach, we first train an AutoGluon model using the 8,000 labeled data points. AutoGluon is an automated machine learning framework that automatically selects the best algorithms and hyperparameters to solve classification problems. We then use this model to predict labels for the unlabeled data, as its relatively high accuracy might help reduce the noise in the predictions. Next, we combine the labeled data with the predicted labels of the unlabeled data to create an enhanced training set.

Finally, we use this augmented training set to train other machine learning models, including K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Logistic Regression (LR), and Random Forests (RF). By incorporating this strategy, we aim to evaluate whether effectively combining labeled and unlabeled data can improve the performance of these models in predicting job salaries.

3.3 Feature Engineering

In this study, we experiment with different feature representations, including embeddings, TF-IDF, and their combination (combined data). Embeddings are dense vector representations that capture semantic meaning, while TF-IDF is a numerical representation that highlights the importance of specific terms. Combining these representations aims to create a more informative feature set for job salary prediction. However, it is worth noting that the performance may not necessarily improve, as the combined data may also introduce noise, potentially affecting the model's accuracy.

3.4 Baseline

As a baseline, we use the K-Nearest Neighbors (KNN) algorithm, which is a simple and intuitive method for classification and regression. It predicts the target value based on the majority vote of the K nearest training instances. We compare the performance of the other machine learning models to the KNN baseline to assess their relative effectiveness.

3.5 Evaluation

In evaluating the algorithms, four metrics are utilized: Accuracy, Precision, Recall Score, and F1 Score. Accuracy represents the ratio of the total number of correct predictions made by the model (Equation 1). Precision is the fraction of correctly classified instances out of the total number of instances that were actually classified (Equation 2). Recall Score, on the other hand, is the fraction of correctly classified instances out of all instances that should have been classified (Equation 3). Lastly, the F1 Score is a composite evaluation metric that combines Precision and Recall Score through a weighted sum (Equation 4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall\_Score = \frac{TP}{TP + FN} \tag{3}$$

$$F1\_Score = 2 \cdot \frac{Precision \cdot Recall\_Score}{Precision + Recall\_Score} \tag{4}$$

To ensure a fair evaluation of the models and prevent overfitting, we use the training set to train the models, the validation set to tune hyperparameters and select the best model, and the test set to assess the final performance of the models. By employing this strategy, we ensure that the evaluation process is robust and reflects the true performance of the models on unseen data.

4 Results

4.1 Embedding

The results in Tables 1 and 2 show that applying semi-supervised learning techniques improves the performance of most models when we use the embedded dataset. For example, the accuracy of the K-Nearest Neighbors model improved from 0.2176 to 0.2366, and the Random Forest model improved from 0.2389 to 0.2447 when including unlabeled data. among these models, the Support Vector Machine showed the highest accuracy in the supervised setting (0.2458), while the Random Forest showed the highest accuracy (0.2447).

Model	Accuracy	Precision	Recall	F1
KNN	0.2176	0.2127	0.2176	0.2112
LR	0.2435	0.2217	0.2435	0.2232
SVM	0.2539	0.2458	0.2539	0.2358
RF	0.2389	0.2343	0.2389	0.2242

Table 1. Performance of Supervised Models

Model	Accuracy	Precision	Recall	F1
KNN	0.2366	0.2255	0.2366	0.2207
LR	0.2401	0.2494	0.2401	0.2050
SVM	0.2429	0.2458	0.2429	0.2099
RF	0.2447	0.2320	0.2447	0.2027

Table 2. Performance of Semi-supervised Models

4.2 TF-IDF

As shown in Tables 3 and 4, the inclusion of unlabeled data through semi-supervised learning techniques resulted in improved performance for all models using the TF-IDF dataset. For example, the accuracy of the K-Nearest Neighbors model improved from 0.1468 to 0.1750, and the accuracy of the Support Vector Machine model improved from 0.2435 to 0.2447 after self-training using unlabeled data. In this dataset, the random forest model performed best in terms of accuracy (0.2458) and precision (0.2288), followed closely by the support vector machine.

Model	Accuracy	Precision	Recall	F1
KNN	0.1468	0.1880	0.1468	0.1361
LR	0.2182	0.2032	0.2182	0.2065
SVM	0.2435	0.2287	0.2435	0.2284
RF	0.2458	0.2288	0.2458	0.2275

Table 3. Performance of Supervised Models

Model	Accuracy	Precision	Recall	F1
KNN	0.1750	0.2051	0.1750	0.1665
LR	0.2245	0.1980	0.2245	0.1965
SVM	0.2447	0.2243	0.2447	0.2149
RF	0.2499	0.2390	0.2499	0.2128

Table 4. Performance of Semi-supervised Models

4.3 Embedding and TF-IDF Combined

The performance of all models also improved when utilizing semi-supervised learning techniques with the combined embedding and TF-IDF dataset, as demonstrated in Tables 5 and 6. The accuracy of the K-Nearest Neighbors model increased from 0.2142 to 0.2251, and the accuracy of the Random Forests model improved from 0.2504 to 0.2545 after incorporating unlabelled data. The combined dataset appears to provide the best performance for most models, such as the Random Forests model, which achieved the highest accuracy of 0.2545. Among the various models, Random Forests and Support Vector Machines consistently displayed the most promising results across different datasets when incorporating unlabelled data through semi-supervised learning techniques.

Model	Accuracy	Precision	Recall	F1
KNN	0.2142	0.2067	0.2142	0.2057
LR	0.2274	0.2160	0.2274	0.2181
SVM	0.2487	0.2362	0.2487	0.2325
RF	0.2504	0.2403	0.2504	0.2375

Table 5. Performance of Supervised Models

5 Discussion

Our findings indicate that incorporating unlabelled data through semi-supervised learning techniques can improve job salary prediction

Model	Accuracy	Precision	Recall	F1
KNN	0.2251	0.2095	0.2251	0.2108
LR	0.2355	0.2194	0.2355	0.2134
SVM	0.2522	0.2402	0.2522	0.2253
RF	0.2545	0.2559	0.2545	0.2180

Table 6. Performance of Semi-supervised Models

performance for various machine learning models. We observed consistent improvements in the models' accuracy, precision, recall, and F1 scores when unlabeled data was included in the training process. This suggests that leveraging unlabelled data effectively can enhance the prediction capabilities of machine learning algorithms.

The results also demonstrate that the Random Forests and Support Vector Machines models generally outperformed other models in terms of accuracy and precision. This can be attributed to their ability to handle high-dimensional data, effectively manage multi-class classification tasks, and capture complex relationships between features and salary classes.

In terms of feature representation, the combination of embeddings and TF-IDF has demonstrated superior performance for the majority of models. This indicates that including a diverse range of feature representations into the feature set can potentially lead to a more comprehensive source of information for job salary prediction. Nonetheless, it is crucial to acknowledge that merging feature representations could also introduce noise, which may have latent effects on the model's accuracy. In this study, the advantages of incorporating extra information seem to surpass the potential downsides, resulting in enhanced predictive performance.

Despite the improvements observed, our study has several limitations. Firstly, we have only explored the self-training approach to semi-supervised learning. Other techniques, such as co-training, label spreading, or label propagation, could be examined to determine their effectiveness in leveraging unlabelled data for job salary prediction. Secondly, we have only investigated four machine learning models. Additional models, including deep learning algorithms and more advanced ensemble techniques, might further improve prediction performance. Lastly, the study relies on a limited dataset, which may not capture the full range of job salary information. Future research could incorporate more extensive and diverse datasets to increase the generalizability of the results.

6 Conclusion

In conclusion, our study demonstrates that effectively combining labelled and unlabelled training data can improve job salary prediction performance for various machine learning models. The use of semi-supervised learning techniques, such as self-training, can help harness the potential of unlabelled data in conjunction with labeled data.

Future research could explore alternative semi-supervised learning techniques and feature representations to further improve job salary prediction performance. Additionally, more sophisticated ensemble learning methods and advanced machine learning models could be employed to combine the strengths of multiple models for enhanced prediction capabilities. Lastly, incorporating larger and more diverse datasets may help improve the generalizability of the findings.

Overall, our findings highlight the importance of leveraging both labeled and unlabelled data in machine learning applications, such as job salary prediction, to maximize the benefits of available data and enhance prediction performance. Despite the limitations of the study, our results provide valuable insights into the potential of semi-supervised learning techniques for improving job salary prediction and other similar applications.

References

Bhola, A., Halder, K., Prasad, A., and Kan, M.-Y. (2020). Retrieving skills from job descriptions: A language model based extreme multi-label classification framework. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5832–5842, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Dutta, S., Halder, A., and Dasgupta, K. (2018). Design of a novel prediction engine for predicting suitable salary for a job. In *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pages 275–279.

Lothe, D., Tiwari, P., Patil, N., Patil, S., and Patil, V. (2021). Salary prediction using machine learning. *INTERNATIONAL JOURNAL*, 6(5).