Salary Prediction via Sectoral Features in Turkey

Şükrü Demir İnan Özer, Berkay Ülke, F. Serhan Daniş and Günce Keziban Orman
Computer Engineering Department, Galatasaray University
İstanbul, TURKEY
inanozer41@hotmail.com, berkayulke99@gmail.com, sdanis@gsu.edu.tr, 0000-0003-0402-8417

2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA) | 978-1-6654-9810-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/INISTA55318.2022.9894130

***Abstract*—Knowing the salary range of a position is beneficial for both job seekers and employers. This work examines the performance of different machine learning methods on salary estimation using industrial variables. The methods are applied to a dataset obtained from Turkey's largest employment platform Kariyer.net. We perform various exploratory analyzes of the data, then use feature engineering techniques for improving the quality of the training data. The effect of the heavy-tailed distribution of salaries is mitigated with various response variable transformations. A timeliness standardization is performed using inflation rates as data from different time periods. Analyses and experiments show that standardization does not have a significant effect on the performance of the model. On the contrary, response variable transformation seems to have a significant effect. As for the models, we conclude that the XGBoost and the artificial neural networks achieve the highest success.**

***Index Terms*—Salary prediction; gradient boosting; machine learning; neural networks; regression**

I. Introduction

In the recruitment processes, the salary range is determined by multiple factors like the requirements for the job, the applied sector and position, the agreement between the employer and the seeker, and most importantly by the current macroeconomic state. For employers, the presence of salary information in the job posting increases both the number of received applications and the percentage of qualified applications among them [1], although Feldman *et al.* [2] states that this only applies to new college graduates, but not to senior executives. Although some parameters of salary determination are obvious, there still exists considerable difference between the salaries of people with similar professional backgrounds that do exactly the same job in the same sector.

In this work, we focus on an objective salary estimation problem that makes use of artificial learning-based intelligent methods. In the recent literature, to predict the salary information of job advertisements in the United Kingdom, a comparison of regression models including Random Forest, Support Vector Machines (SVM), Nearest Centroid, Linear Regression, Logistic Regression, and K-Nearest Neighbor (KNN) Regression is proposed in [3]. Martín *et al.* [4] reports that Random Forest and "ensembles" perform better among several regression and classification algorithms for predicting the salaries of job offers in the information technology sector using sectoral features. Khongchai and Songmuang [5] studies the salary estimation from students' profiles where the past salaries are normalized using linear equations. KNN is reported as the best performant technique against Decision Tree (J48), Naïve Bayes, Multilayer Perceptron, and SVM. In another study, deep learning models are compared to tree-based models like Random Forest and Gradient Boost Trees for the dataset with the features of salary information and personal data of people with bachelor's degrees or higher levels of education [6]. They report that the deep learning models perform faster and better. Jackman and Reid [7] explore the applicability of various regression methods for salary prediction using text descriptions of job postings. Wang *et al.* [8] propose a deep learning model for the prediction of annual salaries by job description. There are other studies that explore the impact of personal features on salary [9]–[12]. These studies still require unknown latent features in order to make more accurate inferences about the salaries. However, some of these required features cannot be used due to the violation of personal data privacy.

The literature lacks studies that use data from Turkey on this subject. Since the economic dynamics might have an effect on the salary determination, the country-basis models create the differences. The aim of this work is to propose an artificial learning-assisted salary estimation model for the jobs in Turkey that is expected to serve both the recruiters and the job seekers. More specifically, we estimate the potential salary range with respect to sectoral differences. An experimental study is conducted on the effects of sectoral features on the salaries of individuals and how salaries can be explained accordingly. Various supervised learning techniques are employed and the performances of the proposed methods are evaluated with several metrics. The salary estimation techniques are applied on the sector-wise datasets obtained from Turkey's largest and oldest online recruitment platform, Kariyer.net. The dataset is compiled with the salary information based on industry and position, excluding rigorously the usage of any personalized information.

The challenges of this work come from the complex nature of the studied dataset. Since the salary distribution is not symmetrical, the model estimations can be faulty. We conduct the necessary operations for these challenges through our contributions. The three contributions of this work are (i) proposing an inflation-corrected timeless salary transformation, (ii) proposing a logarithmic transformation for smoothing the heavy-tail of salary distribution, and (iii) performing different experiments with transformed and raw salaries by applying several state-of-the-art regression algorithms and by using different performance evaluation metrics to build a robust

salary estimation model. The rest of the paper is organized as follows: In Section II, we describe the dataset, the proposed data cleaning and transformation operations, and the regression algorithms with performance evaluation metrics. The results are reported and displayed in Section III. We conclude with the evaluation and discussion of the gained insights in Section IV.

II. Methodology

A. Dataset

The dataset used in this study consists of salary information and sectoral features of a position in a given company. The sample size and the feature dimension of the raw dataset are 500196 and 11 respectively. To comply with the privacy policies, this dataset does not contain any information about companies except the non-descriptive ID numbers and the name of the industry. A small sample from the raw dataset is given in TABLE I. The definitions of the features are listed below:

- **Position Sector**: Industry name under which the position is described with 56 unique values.
- **Position ID**: Identifier of the position with a total of 10170 unique values.
- **Position Name**: The position's full name.
- **Group Code**: Consists of two parts: the collar color of the employee and the level of expertise required for the given position. The collar colors are encoded with B for white-collar workers and M for blue-collar workers. The level of expertise takes an integer value in the 0–4 range.
- **Company ID**: A unique identifier for the company with 53587 values.
- **Salary Type**: The salary type, either gross or net.
- **Leave Date**: The employee's last day of work.
- **Salary Count**: The number of salary payments the employee receives in a year, between 12 and 24.
- **Record Date**: The date the given instance is recorded. The recordings begin on 2018-08-15 and end on 2021-12-23.
- **Salary Amount**: Monthly salary in Turkish Liras (TRY).
- **Company Sector**: The main sector name of the company from 47 unique values.

We implement a basic data cleaning step on this raw dataset. Because the position ID and the leave dates are not relevant for building a salary prediction model, they are discarded in the preparation process. We also discard instances that contain invalid values (e.g. NaN). Moreover, since the group code contains two interrelated features, it is split into two atomic features: the collar color and the level of expertise. After the cleaning process, we obtain 441627 instances with 10 features from the dataset.

1) Data Transformation for Timeliness Standardization: The salaries of the instances in the dataset have two distinct properties: (i) the number of salaries per year and their types (gross or net) can differ from person to person, and (ii) the monthly salaries can be heavily influenced by the macroeconomic facts. Because of these properties, using the raw salary amount feature, which shows the monthly salary, can be misleading. A preliminary transformation setup is used to standardize the salary for all the instances in the dataset. The former property causes a problem in setting a standard for the monthly salaries. We propose two transformation operations. Firstly, a new feature, "annual salary", is introduced by multiplying the monthly salary values by the number of salaries received in a year instead of the basic monthly salary. Secondly, the salary information for all the instances that use gross salary value is transformed into net salaries according to the tax brackets of the related year.

The latter property leads to a salary variation by the date due to Turkey's high inflation rate. High inflation rates cause a significant disparity between the salary of a person with the same qualifications on a specific date and the salary in the preceding year. Thus, we introduce a standardization procedure for the time-dependent salary. For countries where the inflation rate is negligible and the economic factors are stable, such a standardization procedure may not be taken into consideration. However, in the specific regional dataset, the salary information is drawn from the same time frame, for example, the current date, to obtain accurate results. We propose to standardize all annual salaries by the monthly inflation rates of the Turkish Lira from CBRT (Central Bank of the Republic of Turkey) based on the record date feature. The algorithmic procedure of our transformation operation is given in Algorithm 1.

Algorithm 1 Inflation transformation

```
function I(y,m)
    ir ← inflation rate of y^th year's m^th month
    return 1 + (ir / 100)
function F(p, y, m, c)
    if y = c then
        return p
    p ← pI(y, m)
    if m < 12 then
        return F(p, y, m + 1, c)
    else
        return F(p, y + 1, 1, c)
```

Here, the main transformation function $F(.)$ takes four parameters: the salary, p, the year, y, the month of the record date, m, and the current year, c. It first checks if the given salary year is the current year. If not, it simply calculates the cumulative inflation amount and adds it to the given salary in a recursive procedure. The sole inflation rate is computed via an auxiliary function, $I(.)$ which uses the declared inflation rates for each month and year by CBRT. After all the transformations explained in this section, we get a normalized response variable for all cases, regardless of whether the payment is net or gross, the year of the salary, and the number of payments per year.

2) Response Variable Transformation: As demonstrated in the study by Dragulescu and Yakovenko [13], the income distribution follows an exponential distribution for the majority

TABLE I
SAMPLE DATA FROM THE DATASET

Position Sector	Position ID	Position Name	Group Code	Company ID	Salary Type	Leave Date	Salary Count	Record Date	Salary Amount	Company Sector
Sales and Marketing	2043	Overseas Marketing Manager	B2	275080	Gross	20190101	12	2020-12-23 09:53:36.131	8000	IT
Construction	762	Mapping Technician	M2	293103	Net	20190101	12	2021-04-28 12:43:40.675	7500	Construction
Security	4418	Chief of Security	B2	260550	Net	20200101	14	2020-07-21 16:06:58.163	7500	Service
Transportation and Logistics	4326	Terminal Manager	B4	283041	Net	20200101	12	2020-10-08 00:44:33.590	10000	Energy
Finance	4418	Accounting Staff	B1	430737	Net	20200101	12	2020-07-03 16:47:51.306	3250	NaN

of the population. Indeed, when the gain distributions are observed, we see that they are skewed and have long-tailed characteristics [14]. As the regression models assume normal distribution, it is crucial to alleviate the negative effect of non-normal distribution on the models' success. Li *et al.* [3] suggest that applying a logarithmic or square root transformation to skewed data leads to a more symmetrical distribution. Similarly, in another study, logarithmic transformation is applied on continuous variables to preserve the normality [9]. Martín *et al.* [4] also make use of the logarithmic transformation when visualizing a variable with a heavy-tailed distribution. Because of the nature of salary distributions, we perform a logarithmic transformation on our standardized annual salary response variable. These transformations and standardizations serve as the experimental sets for the evaluation of different algorithms' salary modeling performance and for building a robust model for future sector- or position-dependent salary predictions. Hence, we propose three different experimental sets by setting the response variable as: (i) raw monthly salary with the salary count and type in the feature set, (ii) annual salary, and (iii) logarithmic transformed salary.

B. Supervised Learning Methods

1) Regression Algorithms: Since we are to estimate numeric values, the problem is formulated as a regression problem. We employ several regression models: Linear Regression, Bayesian Ridge Regression, Tree-based Models, and Artificial Neural Networks (ANN).

Linear Regression: Despite being one of the simplest supervised learning algorithms, linear regression is powerful in revealing the direct relations between the numerical features and response variables. It estimates the parameter values of a linear equation under the following form for making predictions:

$$y = b_0 + b_1 x_1 + b_2 x_2 ... b_n x_n + \epsilon \tag{1}$$

Here, y is the response variable and x_i is a random variable, i.e. a feature, for explaining y, and ϵ is the error between the predicted value $\hat{y}$ and observed value y. In this study, linear regression serves as a basis for evaluating further models.

Bayesian Ridge Regression: Whereas the famous linear regression approach assumes that the model parameters have unique values that are to be optimized to some point estimates, the Bayesian regression approach, on the other hand, models the regression problem through probability distributions of the parameters. It assumes a prior distribution and returns results for the posterior probability distributions on the model parameters.

Tree-based Models: In tree-based algorithms, the input data is represented by a special tree structure. Each node leads to a sub-tree based on individual features. Each branch linked to a node is a value or a value interval in that node, and each leaf, also referred to as a decision node, represents a decision of the model. The most significant benefit of using a tree-based model is that the generated model is fully explainable to humans because each node in a tree can be thought of as a question asked to the data, and each leaf is the prediction for the given answers. By their nature, tree-based models excel with data that mostly have categorical features. Each class of a categorical feature can be used by a node as a criteria, while for numerical features, models need to find certain thresholds, which can be hard to optimize. For these reasons, decision tree, random forest, and XGBoost models are employed in our experiments.

Artificial Neural Networks: ANNs are formed of primitive nodes of linear computation units connected to each other with weights that are to be tuned in the process of training. Because the primitive nodes, or neurons, are also exposed to nonlinear activation functions, ANNs are capable of fitting complex functions. However, they expect to receive a larger number of instances than the other machine learning methods. As our specific dataset contains half a million instances, the ANN models are positioned to be highly appropriate for our study. Moreover, the ANN can standardize the response variable since the year and salary type information are used along with the salary amount of each instance. Thus, it can be useful for revealing the effects of the transformations we proposed in the previous section.

C. Performance Evaluation Metrics

To evaluate the performance of models, we present four different metrics: Root Mean Squared Error (RMSE) repre-

sents the standard deviation of the prediction errors between predicted and observed values of response variable (2).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2}$$

Mean Absolute Error (MAE), given in (3), is the average of measured absolute errors.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3}$$

Mean Absolute Percentage Error (MAPE) measures the accuracy percentage of the prediction (4).

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{4}$$

Coefficient of Determination (R Squared or R^2) is a measure of how much of the variations in the predictions made can be explained by the model (5).

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \tag{5}$$

where $SS_{res} = \sum_i (y_i - \hat{y}_i)^2$ and $SS_{tot} = \sum_i (y_i - \overline{y})^2$ with y is the vector of real values from observed data and $\hat{y}$ is the vector of predicted values and $\overline{y}$ is the average of the predicted values.

D. Experiments

All the experiments are performed with Python using several different libraries. The linear regression, Bayesian regression, decision tree, and random forest experiments are designed and performed with the scikit-learn library [15]. The XGBoost library [16] is used for implementing the XGBoost algorithm. For tuning the hyper-parameters of the aforementioned models, scikit-learn's grid search functions are applied with 5-fold cross validation. Experiments with ANN models are performed with TensorFlow [17] and the Keras API [18]. Trials are conducted with an empirical approach in hyper-parameter selection for ANN models. In these trials, the number of hidden layers varies between 2 and 20. For the number of nodes in each layer, powers of two from 8 to 512 are used. The best results are obtained with 10 hidden layers and 256 nodes in each layer. Stochastic Gradient Descent, RMSprop, and Adam algorithms are used as optimization functions. As the activation functions of the layers, sigmoid, tanh, ReLU, ELU, and PReLU activation functions are used. For the weight initialization methods, Xavier uniform initialization is used for layers with tanh and sigmoid activation functions, and He uniform initialization is used for layers with ReLU, PReLU, and ELU activation functions. Experiments are conducted with the Huber loss as the loss function, considering that it would reduce the effect of very high values in the salary data since it is less sensitive to outliers.

While performing the experiments, 20% of the dataset (88326 instances) is spared as test data. In the experiments with ANN method, 20% of the remaining data is used as the validation data, and the rest for training the model.

III. Results

A. Data Descriptive Analysis

After the cleaning steps that is explained in section II-A, white-collar employees account for 76% of the instances, while blue-collar employees account for only 24%. A descriptive histogram plot of both groups before logarithmic transformation is shown in Fig. 1 on the left. Salaries from our dataset displayed a similar skewed distribution as expected in section II-A2. The distribution after the transformation can also be seen in Fig. 1 on the right. We obtain a wider, less skewed, and more normal-like shape, but still, the salary distribution is not symmetric even after the logarithmic transformation.

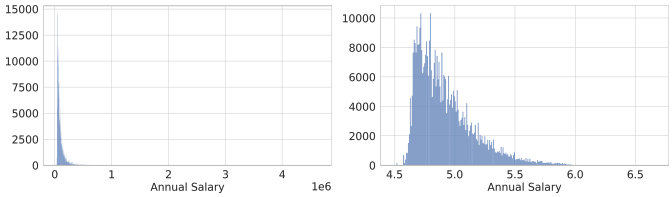

Fig. 1. Annual Salary frequency graph before (left) and after (right) $log(x)$ transformation

In Fig. 2, the positive relationship between the level of expertise and the salary received can be seen. The skewed distribution of the salary feature causes outliers to appear at higher salary ranges for each expertise level, which is for some extend is solved by the logarithmic transformation on salary feature.

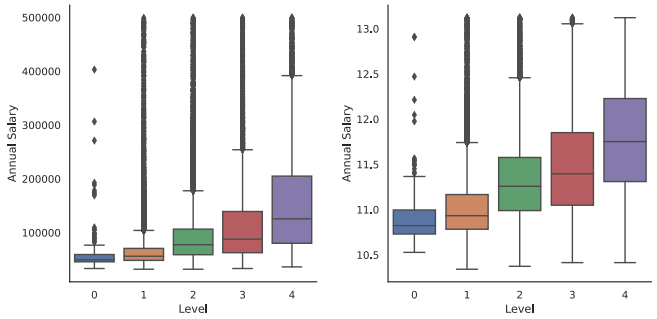

Fig. 2. Boxplot of expertise level and annual salary (limited to 5e5 TRY) before (left) and after (right) logarithmic transformation

B. Supervised Learning Results

The test performance result of the first experiment on the dataset with monthly salary values as a response variable is displayed in TABLE II. Accordingly, the best performance is obtained by the XGBoost algorithm according to R^2, RMSE, and MAE. The lowest MAPE is obtained by the ANN. The decision tree has the highest error and lowest R^2.

We can observe the corresponding residual plot of the best performance for this experiment set in Fig. 3 on the left. A bias of the model prediction can be observed in this plot since the residual plot has a shape and does not seem random or

TABLE II
PERFORMANCE EVALUATIONS WITH ORIGINAL MONTHLY SALARY

Model	R^2	RMSE	MAE	MAPE
Linear Regression	0.23	4278	2286	43.69%
Bayesian Ridge	0.23	4300	2282	43.67%
Decision Tree	0.13	4630	2255	39.86%
Random Forest	0.23	4314	2144	38.04%
XGBoost	**0.31**	**4098**	**2075**	38.37%
ANN	0.29	4163	**2091**	**37.87%**

uniform. Since the error rates are high, R^2 is low, and residuals are not uniformly distributed, the use of the original monthly salary is not seen as a good response variable.

One can observe the results of the second experiment in TABLE III for the dataset with annual salary after standardization. Again, the XGBoost and ANN algorithms appear to be one step ahead and decision tree is one step behind of the other algorithms.

TABLE III
PERFORMANCE EVALUATIONS WITH STANDARDIZED ANNUAL SALARY

Model	R^2	RMSE	MAE	MAPE
Linear Regression	0.24	77769	42553	43.32%
Bayesian Ridge	0.24	77772	42523	43.27%
Decision Tree	0.19	79916	40125	38.63%
Random Forest	0.24	79456	39441	37.70%
XGBoost	**0.31**	**74146**	**38607**	38.17%
ANN	0.29	77004	**38785**	**37.22%**

Although using standardized annual salaries improves algorithm success ($\Delta\overline{R^2} = 0.01389$), it is not as effective as expected. From these results, we can deduce that all the algorithms are capable of modelling annual salaries by taking into account the effect of inflation on salaries. The corresponding residuals of the best performance of this experiment can be seen in Fig. 3 in the center. Like the previous one, the residual plot exhibits a visible shape.

The results of the third experiment are represented in TABLE IV. Like in two previous experiments, XGBoost and ANN showed the best performance, while the decision tree showed the worst one. However, this time, decision tree performance is close to other algorithms' while it was far beyond in previous experiments. When TABLE III and TABLE IV are examined together, the logarithmic transformation of the response variable has a positive and significant effect on success, increasing it nearly by 50% ($\Delta\overline{R^2} = 0.14265$).

The residuals for the XGBoost predicted values for this experiment is shown in Fig. 3 on the right. It can be deduced that a logarithmic transformation reduces heteroscedasticity to a more preferable distribution.

XGBoost and ANN models show higher mean R^2 scores compared to other methods in every experiment, difference diminishing with improvements made on the training data ($\Delta R^2 = 0.0956$, $\Delta R^2 = 0.0666$ and $\Delta R^2 = 0.0485$). Despite the fact that the success metrics of those models were close to each other, the residual graphs of the XGBoost and ANN models revealed similar but different patterns that can be seen in Fig. 3 on the right and Fig. 4.

TABLE IV
PERFORMANCE EVALUATIONS WITH TRANSFORMED STANDARDIZED ANNUAL SALARY

Model	R^2	RMSE	MAE	MAPE
Linear Regression	0.37	7.53e-3	5.68e-3	2.85%
Bayesian Ridge	0.38	7.48e-3	5.64e-3	2.83%
Decision Tree	0.37	7.55e-3	5.50e-3	2.75%
Random Forest	0.40	7.39e-3	5.41e-3	2.71%
XGBoost	**0.43**	**7.16e-3**	**5.30e-3**	**2.66%**
ANN	**0.43**	**7.20e-3**	**5.32e-3**	**2.67%**

C. Discussion

Considering all three experiments together for comparing the algorithm performances, the Linear Regression and Bayesian Ridge methods show relatively moderate success in the first experiments and perform better after the preprocessing steps. As linear regression explains the data with a mean value, the high variance of the data adversely affects its success. It can be said that the decision tree model explains a subset of the data better, but it is not as successful in explaining the data as a whole. The decision tree approach is the most beneficial after the training data improvements, increasing its R^2 score by approximately 200% ($\Delta R^2 = 0.2422$). In all experiments, XGBoost and ANN consistently yield the best results. This can be interpreted as that these models have the ability to make successful inferences even from noisy and unprocessed data. In other words, it can be said that those models have the ability to optimize the parameters so that salary can be standardized, and even standardize it implicitly better than the transformation processes.

The residual plots reveal that models tend to overestimate high salary values and underestimate low ones, despite the improvement attempts made on the data. It is possible to deduce that there are latent variables which may affect the salary values, but those variables are not present during the training of the models. It is also possible that we require more sophisticated feature engineering methods before repeating the experiments.

IV. CONCLUSION

In this work, we build a machine learning task for predicting the salary ranges objectively for the sector-based data taken from Kariyer.net in Turkey. While the main challenges are due to the heterogeneity, dirtiness, and complex nature of the studied data, we also tackle the traditional data cleaning tasks on the sector-based data features with more sophisticated data transformation tasks for salary amounts. We propose two transformations: a timeless annual salary correction by the inflation rate instead of the raw monthly salary and a logarithmic transformation to overcome the heavily-tailed distributed data, which causes a problem for most of the modelling tools according to the literature. The experiments are performed with six famous state-of-the-art regression algorithms. The

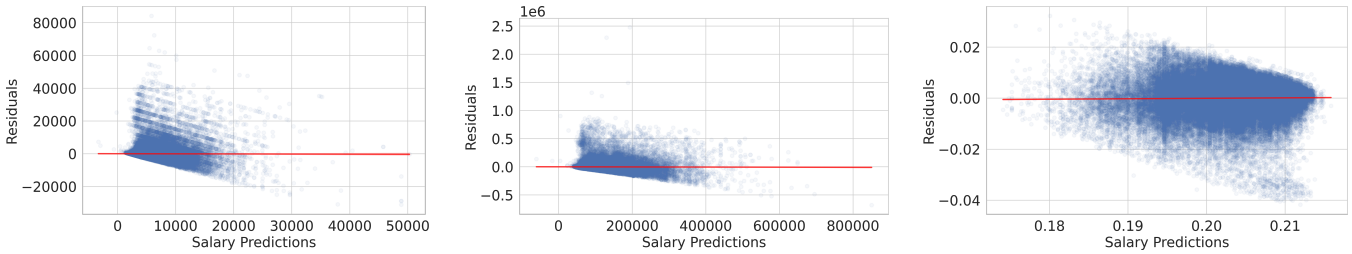

Fig. 3. Residual plots of XGBoost for first (left), second (center) and third (right) experiments

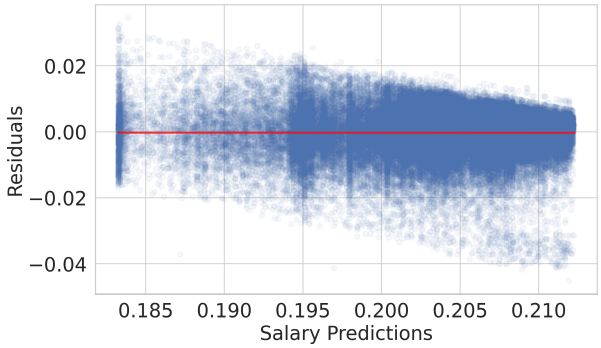

Fig. 4. Residual plot (ANN) - After logarithmic transformation

two most notable findings are that (i) recalculating salaries using the inflation rate has no significant effect and (ii) using logarithmic transformation has a positive effect on the salary estimation. Besides, among all the tested algorithms, XGBoost and ANN models stand out as the most successful predictors without being affected by the effect of inflation on salaries.

There are improvements to be made on the data preprocessing step. Explicitly speaking, deletion of incomplete instances may lead to the loss of valuable information. Techniques such as multiple imputation allow said cases to be used. It may also be interesting to repeat the study with data from different sectors in a more comprehensive dataset. Lastly, sector profiles in the data can be determined instead of training a single model to cover all sectors, and different forecasting models can be constructed for each profile.

ACKNOWLEDGMENT

We thank Kariyer.net Elektronik Yayıncılık ve İletişim Hizmetleri A.Ş for providing Salary dataset. This article is partially supported by Galatasaray University Research Fund (BAP) within the scope of project number fba-2021-1063, and titled ”Niteliklendirilmiş çift yönlü ağlarda bağlantı tahmini ile öneri sistemleri geliştirilmesi”.

REFERENCES

[1] A. B. Kaplan, M. G. Aamodt, and D. Wilk, “The relationship between advertisement variables and applicant responses to newspaper recruitment advertisements,” *Journal of Business and Psychology*, vol. 5, no. 3, pp. 383–395, 1991. [Online]. Available: http://www.jstor.org/stable/25092294

[2] D. C. Feldman, W. O. Bearden, and D. M. Hardesty, “Varying the content of job advertisements: The effects of message specificity,” *Journal of Advertising*, vol. 35, no. 1, pp. 123–141, 2006. [Online]. Available: http://www.jstor.org/stable/20460716

[3] L. Li, X. Liu, and Y. Zhou, “Prediction of salary in uk,” 2013.

[4] I. Martín, A. Mariello, R. Battiti, and J. A. Hernández, “Salary prediction in the it job market with few high-dimensional samples: A spanish case study,” *International Journal of Computational Intelligence Systems*, vol. 11, pp. 1192–1209, 2018.

[5] P. Khongchai and P. Songmuang, “Implement of salary prediction system to improve student motivation using data mining technique,” in *2016 11th International Conference on Knowledge, Information and Creativity Support Systems (KICSS)*, 2016, pp. 1–6.

[6] P. Viroonluecha and T. Kaewkiriya, “Salary predictor system for thailand labour workforce using deep learning,” in *2018 18th International Symposium on Communications and Information Technologies (ISCIT)*, 2018, pp. 473–478.

[7] S. Jackman and G. Reid, “Predicting job salaries from text descriptions,” Apr 2013.

[8] Z. Wang, S. Sugaya, and D. P. Nguyen, “Salary prediction using bidirectional-gru-cnn model,” *Assoc. Nat. Lang. Process*, 2019.

[9] K. R. Hamlen and W. Hamlen, “Faculty salary as a predictor of student outgoing salaries from mba programs,” *Journal of Education for Business*, vol. 91, pp. 38 – 44, 2016.

[10] J. Loeb and M. Ferber, “Sex as predictive of salary and status on a university faculty,” *Journal of Educational Measurement*, vol. 8, no. 4, pp. 235–244, 1971.

[11] C. B. Johnson, M. L. Riggs, and R. G. Downey, “Fun with numbers: Alternative models for predicting salary levels,” *Research in Higher Education*, vol. 27, no. 4, pp. 349–362, 1987.

[12] J. Mainert, C. Niepel, K. Murphy, and S. Greiff, “The incremental contribution of complex problem-solving skills to the prediction of job level, job complexity, and salary,” *Journal of Business and Psychology*, vol. 34, 12 2019.

[13] A. Drăgulescu and V. M. Yakovenko, “Exponential and power-law probability distributions of wealth and income in the united kingdom and the united states,” *Physica A: Statistical Mechanics and its Applications*, vol. 299, no. 1, pp. 213–221, 2001, application of Physics in Economic Modelling.

[14] D. Neal and S. Rosen, “Chapter 7 theories of the distribution of earnings,” in *Handbook of Income Distribution*. Elsevier, 2000, vol. 1, pp. 379–427.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[16] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[17] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, “Tensorflow: A system for large-scale machine learning,” in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[18] F. Chollet *et al.*, “Keras,” https://keras.io, 2015.