

# Key Factors to Consider when Predicting the Costs of Forwarding Contracts

Quang Hieu Vu  
 ZaloPay, VNG Corporation, Vietnam  
 hieuvq@vng.com.vn

Ling Cen, Dymitr Ruta, Ming Liu  
 EBTIC, Khalifa University, UAE  
 {cen.ling,dymitr.ruta,liu.ming}@ku.ac.ae

**Abstract**—Predicting the cost of forwarding contracts is a typical problem that logistics companies need to solve in order to optimize their business for a better profit. This is the challenge defined in the FedCSIS 2022 Competition where a five-year history of contract data and their delivery routes from a large Polish logistics company are provided to train a Machine Learning model. In addition to the contract data, historical wholesale fuel prices and euro exchange rates at the contract time are also provided. To address this challenge, we first designed a basic solution where we focused on feature engineering to find good impact features for the model. After that, the same set of features were used to train two different models: one using XGBoost and the other using LightGBM. The average predictions of the two boosting models were then used as the predictions for the next post-processing step. Finally, in the post-processing step, we designed and trained a simple linear regression model to capture the average monthly changes of the contract cost, given the changes of the fuel prices and euro exchange rates. These captured changes were used to post-process (adjust) the predictions in the previous step to address the issue that tree-based models could not predict the value that they did not see before. While the basic solution with careful feature selection gave us a place in the top-5, our post-processing strategy in the last step helped us win the 3<sup>rd</sup> prize in the competition.

**Index Terms**—Logistics, Forwarding Contract Cost Prediction, Gradient Boosting Trees, XGBoost, LightGBM, Linear Regression, Feature Engineering, Post-Processing.

## I. INTRODUCTION

MANY logistics companies are on the road to digital transformation and employ AI/Machine Learning technologies to support and optimize their daily business. One of the key challenges that these companies are facing is to predict to cost of a delivery/forwarding contract in which an accurate prediction result can help the logistics companies in many aspects of which three typical ones are listed below:

- To maximize revenue by selecting profitable contracts.
- To identify issues and optimize operations to reduce costs.
- To get better planning in contract execution.

The challenge of predicting the cost of forward contract is exactly the objective of the FedCSIS 2022 [1] competition<sup>1</sup>, which is in cooperation with PTI and QED Software and sponsored by Control System Software<sup>2</sup> – a software company that has been delivering solutions for the Transportation, Spedition, and Logistics industry for 20 years. In this competition, a

five-year history of contract data and their delivery routes together with wholesale fuel prices and euro exchange rates at the contract time are provided. Specifically, two types of data sources are provided. While the first data source contains basic information about the contracts, the second one describes the main sections of the planned routes associated with each contract. Given the data, when we performed exploration data analysis (EDA), we could see that the amount of data is not big, there are not much information in the contract to analyse, and the number of provided features in the contract data is small. As a result, we believed that the most important factor to design and train a good prediction model is in the feature engineering process where (1) We need to generate extra features to be used, in addition to the existing provided features and (2) useful features should be carefully selected for the model. With respect to this point, our first contribution in this paper is in this feature engineering process. Once we got the selected features, we then simply trained two boosting tree models: XGBoost and LightGBM and obtained the average predictions from the two models as the forecast.

Our second contribution in this paper is a simple and effective approach to post-process the predictions to capture the trends in contract costs. Basically, it is well-known that using tree-based models in the presence of trends over time can lead to inaccurate results. The reason is in the way tree-based models make predictions. For example, Decision Trees make regression predictions by seeing which “leaf” the data point belongs to and assigning the average of the target variable from the training set to that point. In this case, they fail to accurately predict values they have not already seen, and values from a significantly different population (perhaps after some trend in time) will cause the model to make inaccurate predictions. Random Forests and Gradient Boosting Tree algorithms suffer the same problem because their results are averages results of Decision Trees. In this competition, when we performed EDA, we could see the fuel prices have steadily increased in the past couple of years. Euro exchange rates have also increased during the time of the test set. As a result, there should be an increase trend in the average cost of the contracts in the test set, which may not be well captured by the models in the previous step. Therefore, we designed and trained a simple linear regression model to capture the trend in the average cost of forwarding contracts and use this model’s result to post-process the predictions from tree-based models. This post-

<sup>1</sup><https://knowledgepit.ml/fedcsis-2022-challenge/>

<sup>2</sup><https://controlsystem.com.pl/>

processing step did help to uplift our prediction model and brought us to the 3rd position in the final ranking list.

To summarize, our contributions are twofold:

- We present our feature engineering process in which we first share how to generate extra features using route information and then select good features having impacts to the forecast model from a set of candidate features.
- We introduce a simple linear regression model used in the post-processing step to overcome an issue of tree-based models in capturing trends in contract costs.

For the rest of the paper, we organize the content as follows. The background and related work are introduced in Section II. An overview of the competition challenge is described in Section III. The details of the proposed method are presented in two sections, where Section IV is dedicated for feature engineering and feature selection while Section V shows details of the model design, focusing on the linear regression model used in the post-processing step. Finally, conclusions are given in Section VI.

## II. RELATED WORK

The costs of forwarding contracts can be affected by many factors, which, besides contract nature and transportation arrangement, fuel prices, currency exchange trends are decisive too. The actual transportation cost is also constrained by some factors such like behavior of drivers, weather, traffic, market demand, etc., accurately forecasting the cost is, thus, challenging, but critical in the business providing logistics service or involving supply chains [5].

Artificial intelligence techniques empowered solutions have been investigated to solve transportation problems in both industry and academy. A logit neural network (NN) based mode selection model was developed to address border transportation in [11]. Freight demand was predicted for inter-modal terminals by NNs in [12]. Traffic volume in non-urban highways under heterogeneous conditions and vehicle counts were predicted by developing a NNs based model in [13].

There is, however, quite little work published in the literature on cost prediction of forwarding contracts. The work in [14] developed AI based models to predict the long-term cost of the logistics service, and attempted to construct a risk-aware interval for the prices to be offered in the bid, aiming to boost competitiveness in the application for tenders, and in addition, historical data was used to develop statistical learning models for predicting the success likelihood of a tender based on the actual data and predicted service prices achieved from previous stage. In [5], a trapezoidal neutrosophic fuzzy analytical hierarchy process (TNF-AHP) was proposed to determine the most significant criteria that were used to predict transportation cost by an artificial neural network (ANN) model, which, claimed by the authors, can be also used in supply chain management and inventory control management.

Boosting is a popular technique used in machine learning to reduce errors in predictions from which to improve the accuracy of machine learning models. The basic idea of boosting approach is to combine a set of weak models into a strong

and robust model, which is able to reduce the prediction bias [6], [7]. Gradient boosting (GB) is an extension of boosting, in which the process of additive generation of weak models is based on a gradient descent algorithm over an objective function [8]. Gradient boosting decision tree (GBDT) is an ensemble model of decision trees, used as weak learners in the gradient boosting ensemble model. In each iteration of its training process, a new decision tree is added to the pool, which tries to increasingly learn on mistakes of decision trees already in the pool by fitting the negative gradients (or residual errors) [9], [10].

## III. COMPETITION DESCRIPTION

The challenge of the FedCSIS 2022 [1] is to predict the cost of forward contracts, using five-year history of contract data and their delivery routes together with wholesale fuel prices and euro exchange rates at the contract time. The data is provided in three different files as follows:

- Main contract data: contain general information of forward contracts that are ready to be used as features to train a model. The data is stored in two .csv files: “css\_main\_training.csv” and “css\_main\_test.csv”.
- Contract route data: provide detailed information about routes of forward contracts, which can be used to generate extra features (note that some basic route features are already created in the main contract data). The data is stored in two .csv files: “css\_routes\_training.csv” and “css\_routes\_test.csv”.
- Wholesales fuel prices: store average fuel prices at monthly level throughout the time of forward contracts in a single .csv file, “fuel\_prices.csv”.

The accuracy of the prediction model is measured by the Root Mean Square Error (RMSE) metric. Note that the purpose of using RMSE is to give higher penalty for large errors compared to smaller ones because the errors are squared before they are averaged.

## IV. FEATURE ENGINEERING

The process of feature engineering, which typically includes feature generation at first, followed by feature selection using Greedy Forward Search, Greedy Backward Search, and finally SHAP analysis [15] for feature importance, is clearly presented in this paper [16]. Thus, in this section, we simply discuss which sources we used to generate our features as well as the list of our selected features.

In general, the features that were used to train our models to predict the cost of forward contracts were generated in the two basic steps. In the first step, we executed “feature generation” during which extra features are generated from the first two data sources: “main training data” and “contract route data”. After that, we performed “feature selection” to determine the usefulness of generated features based on their impact on the performance of our models and finally chose which are the features that should be included in training our final models. In general, our features could be classified into “basic features” which are available in the “main contract

data” and “extra features” which are generated in the feature generation process.

#### A. Basic features

We use all available features in the “main training data” but one and they are all good features in the top-20 important features returned by our model. The only exception is from the “euro\_exchange\_rate”, which we thought to be an important feature, but the model told us otherwise. We tried a number of ways to utilize the “euro\_exchange\_rate” such as using it as a single feature, identifying trend up or down in the exchange rates, or converting it into a mean-encoding feature. But, none of the above approaches was succeeded. As a result, we did not use the “euro\_exchange\_rate” in the main model. However, it turned out that this feature was still useful in the linear regression model that we employed later in our post-processing step presented later in Section V.

#### B. Extra features

Extra features are mainly generated from “css\_routes” files. These features help to identify special properties of the routes from which giving the model better forecast accuracy. They include the following sets of features:

- The route statistics features: include the number of route segments, the number of route segments with and without cargo, and the number of route segments where starting and ending points are in the same country (or in different countries). These features give us extra information on how big or complexity the forward contract is, in addition to the existing “km distance” feature.
- The gap feature between “km distance” and “Haversine distance”: this feature provides us how easy or hard a contract forward can be executed. Picture that if the “Haversine distance” is short while the “km distance” is long, the implication is that it is not easy or straightforward to move directly from the start to the end of the contract as some detour may be required in the trip.
- The statistics of vehicles and trailers used in the trip: include the average of axle counts from all vehicles and trailers, the average of kerb weight from all vehicles and trailers, the average of vehicle engine capacity and the average of trailer payload. These features tell us how heavy the cargo is in the trip, in addition to the “max weight” feature in the main contract data.
- The features about distance between the route start and the point where the cargo is loaded first as well as distance between the route end and the point where the cargo is last unloaded. These features tell us the percent utilization of vehicles in the contract.

In addition to route features, as we have a text describing the temperature requirement in the main contract data, we generated the following temperature requirement features, which are good features for the model because the forward contract cost should be higher if there are special requirements for the temperature such as frozen or automatic. Specifically, these features include:

- The range of temperature (low and high) if they are mentioned in the text field. Otherwise, they are left with NULL values. In some cases, where a fixed temperature is required, we set equal values for both the low temperature and high temperature.
- One-hot encoding features for the top-5 important words detected from the text such as frozen, continuous, or automatic. These words are created simply by getting the top-5 words returned from “CountVectorizer” after removing stop words.

Finally, as we have the time when the trip starts and ends in the main contract data, we generate these following extra time related features:

- The weekdays where the trip starts and ends as the cost could be different if we start or end the trip at different days of the week. For example, the trip starts or ends during the weekend may lead to a higher contract cost.
- Similar to the above case, the hour of day where the trip starts and ends also have impacts on the contract cost. A night time start or end is expected to have a higher cost compared to other time of the day.
- Finally, the day of month where the trip starts and ends and the duration (in terms of hours and days) of the trip also contribute to changes in the contract cost.

#### C. Model design and implementation

Given the list of good features obtained in the above feature engineering step, we simply designed and trained two boosting trees: XGBoost and LightGBM, using the same set of features and the final prediction is the average predictions returned from the two models. As you can expect, the combination of the two models using the same set of features does not help to make much improvement in the model accuracy. Instead, we employ this strategy simply to get a stable prediction result as our main concern is always the overfitting issue given the small amount of data used for the public leaderboard, which we experienced in the past two competitions of 2020 [2], [3] and was presented in the paper [4].

### V. FORECAST POST-PROCESSING WITH LINEAR REGRESSION

As discussed in Section IV, we could not utilize the euro exchange rates in the main model. Similarly, when we tried to add wholesale fuel prices to the main model, it does not have a positive impact. However, when we looked at the changes in both euro exchange rates and fuel prices, we could see that during the period of testing, there was a significant increase in both euro exchange rates and fuel prices. As discussed in Section I, since using tree-based models in the presence of trends over time can lead to inaccurate results, this section introduces our solution to address this issue. Specifically, we will present a design and implementation of a simple linear regression model to predict monthly changes in the average contract costs in Section V-A and then use the predicted result to adjust the predictions returned by the main model in Section V-B.

### A. Forecasting trend in contract costs

To detect trend in contract costs, we built a simple model to detect the correlation between changes in euro exchange rates and fuel prices and changes in the average contract cost at the monthly level. Note that here we assume that the types of executing contracts each month follow a similar distribution. As the impact of euro exchange rates and fuel prices could be different given different trip distance (e.g., short distance trip may be more or less sensitive to the euro exchange rates and fuel price changes compared to long distance trip) and whether train routes or ferry routes are involved in the trip, we need to consider this factor into the model. In the end, we trained a linear regression model using the following features aggregated at the monthly level:

- The minimum, mean, and maximum of euro exchange rates as well as fuel prices.
- The distance group which we defined based on the total km of the trip. In our solution, we split contracts equally into 4 groups having total km greater than 820, between 430 and 820, between 230 and 430, and less than 230.
- The last features are indicators of whether the trip has train routes or ferry routes.

The training target of the model is the average contract costs each month obtained from the training data set. Once the model is trained, we use it to predict the average contract costs for months in the test data set.

### B. Post-processing predictions

Given the predicted average contract cost of a month returned from the above linear regression model (let's call it  $A$ ), we compare it against the average predicted contract costs returned from our main model trained with boosting trees presented in Section IV (let's call it  $B$ ), three cases may happen:

- If  $A$  is equal to  $B$ , our two prediction models are aligned. It is good and we should not do anything for post-processing.
- In cases  $A$  is less than  $B$ , there could be a down trend in the average contract costs that fail to be captured by our tree-based models, and hence the models generate over-forecast. In this case, we first compute  $\delta = B - A$  and then subtract  $\delta$  from all predictions returned from the main model.
- In cases  $A$  is greater than  $B$ , it is opposite as there could be an up trend that the tree-based models fail to capture, and hence we need to add in a  $\delta = A - B$  for all predictions made by the main model.

It is interesting to note that in our case, the linear regression model always returned a higher prediction for the average contract costs. It means that there should be an expected increase in the contract costs, given the hike of the euro exchange rates and the fuel prices during the period of time used for testing. In our solution, by applying this post-processing step, we could see a score improvement of 0.006 in the Public Leader Board,

which is a good improvement, given that even good features could only help to improve score of 0.002 to 0.003.

## VI. CONCLUSIONS

In this paper, we presented our solution to win the 3<sup>rd</sup> prize of the FedCSIS 2022 Challenge. There are two key factors leading to the effectiveness of our solution. The first one is a process of feature engineering to generate extra useful features and then carefully select features for the model. The second one is a solution for post-processing the predictions in order to capture changing trends in the forward contract costs, which otherwise are failed to get in the main tree-based models. They were both discussed in the paper.

## REFERENCES

- [1] A. Janusz, A. Jamiolkowski, M. Okulewicz, "Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results", *Proceedings of the 17th Conference on Computer Science and Intelligent Systems (FedCSIS)*, 2022.
- [2] A. Janusz, M. Przyborowski, P. Biczuk, D. Ślęzak, "Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit", *Proceedings of the 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, doi: 10.15439/2020F159.
- [3] A. Janusz, G. Hao, D. Kaluza, T. Li, R. Wojciechowski, D. Ślęzak, "Predicting Escalations in Customer Support: Analysis of Data Mining Challenge Results", *IEEE International Conference on Big Data*, 2020, doi: 10.1109/BigData50022.2020.9378024.
- [4] D. Ruta, L. Cen, Q. H. Vu, "Deep Bi-Directional LSTM Networks for Device Workload Forecasting", *Proceedings of the 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, doi: 10.15439/2020F213.
- [5] A. Singh, A. Das, U. K. Bera, G. M. Lee, "Prediction of Transportation Costs Using Trapezoidal Neutrosophic Fuzzy Analytic Hierarchy Process and Artificial Neural Networks", *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3098657.
- [6] L. Breiman, "Bagging predictors", *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
- [7] L. Mason, J. Baxter, P. Bartlett, and M. Frean, "Boosting Algorithms as Gradient Descent", in S.A. Solla, T.K. Leen, and K.R. Muller, editors, *Advances in Neural Information Processing Systems 12*, MIT Press.
- [8] L. Mason, J. Baxter, P.L. Bartlett, and M. Frean, "Boosting algorithms as gradient descent", *Proceedings of International Conference on Neural Information Processing Systems*, MIT Press, 1999.
- [9] J. H. Friedman, "Greedy function approximation: A gradient boosting machine", *Ann. Statist.*, vol. 29, no. 5, pp. 1189-1232, 2001.
- [10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", *Proc. Neural Information Processing Systems Conference (NIPS)*, 2017.
- [11] U. Gazder and N. T. Ratrouf, "A new logit-artificial neural network ensemble for mode choice modeling: A case study for border transport", *J. Adv. Transp.*, vol. 49, no. 8, pp. 855-866, 2015.
- [12] I. C. Bilegan, T. G. Crainic, and M. Gendreau, "Forecasting freight demand at intermodal terminals using neural networks—an integrated framework", *Eur. J. Oper. Res.*, vol. 13, no. 1, pp. 22-36, 2008.
- [13] K. Kumar, M. Parida, and V. K. Katiyar, "Short term traffic flow prediction for a non urban highway using artificial neural network", *Procedia - Social and Behavioral Sciences*, vol. 104, pp. 755-764, 2013.
- [14] S. Nataraj, C. Alvarez, L. Sadaa, A. Juana, J. Panadero, C. Bayliss, "Applying Statistical Learning Methods for Forecasting Prices and Enhancing the Probability of Success in Logistics Tenders", *Transportation Research Procedia (Elsevier)*, vol. 47, 2020.
- [15] S. Lundberg, S. Lee, "A Unified Approach to Interpreting Model Predictions", *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.
- [16] Q. H. Vu, D. Ruta, L. Cen, M. Liu, "A combination of general and specific models to predict victories in video games", *IEEE International Conference on Big Data (Big Data)*, 2021, doi: 10.1109/Big-Data52589.2021.9671285.