Machine Learning Approaches for Forecasting Electronic Fund Transfer Patterns in Private Banking

by

Mehmet Kubilay Karatopcu

A Project Report Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Data Science

KOÇ
ÜNİVERSİTESİ

August 14, 2024

Machine Learning Approaches for Forecasting Electronic Fund Transfer Patterns in Private Banking

Koç University
Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's project by

Mehmet Kubilay Karatopcu

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

____

Assoc. Prof. Özden Gür Ali (Advisor)

Date: ______________________

ABSTRACT

Machine Learning Approaches for Forecasting Electronic Fund Transfer Patterns in Private Banking

Mehmet Kubilay Karatopcu

Master of Science in Data Science

August 14, 2024

Electronic Fund Transfers are an important part of that effects liquidity of a bank and predicting it can help businesses drive their decisions more efficiently and accurately. This study explores the application of machine learning techniques to predict electronic fund transfer (EFT) patterns in the private banking sector. It focus on two primary transfer types which are EFT (outbank) and 'Havale' (in-bank transfers) and aims to forecast both the timing and type of future transactions of customers. The predictive models categorizes future transactions into four time frames: tomorrow, within 3 days, within 14 days, or more than 14 days from the last transaction, and alternatively within 14 days or higher. Additionally, it predicts whether the next transaction will be an outbank or a in-bank transfer.For the models, XGBoost, Random Forest, Logistic Regression, and LSTM networks are trained on historical transaction data with various features that may influence transfer patterns. By accurately forecasting EFT patterns, this study aims to provide better understanding of customer behavior, optimize resource allocation, and potentially detect anomalous activities, thus strengthening overall banking operations and security.

Chapter 1:
INTRODUCTION

In the fast evolving financial technology world, electronic fund transfers (EFTs) have become an important part of modern banking operations. The efficiency and reliability of EFT systems are crucial for the smooth functioning of financial institutions. In Turkey, the development and evolution of the EFT system provide a good objective for our research into predicting electronic fund transfer patterns in private banking.

“The EFT System provides real time transfer and real time gross settlement of Turkish Lira interbank payments.” (Central Bank of the Republic of Turkey, n.d.) Thus, in this fast paced fund transfer mechanism banks should adjust their strategies dynamically.And due to EFT infrastructure millions of electronic fund transfers are executed daily for both in-bank (havale) and out-bank (EFT). The volume and variety of these transactions generate vast amounts of data, presenting both challenges and opportunities for financial institutions.

In our context, we focus on predicting EFT patterns via machine learning in a private bank. By analyzing historical transaction data and applying algorithms such as XGBoost, Random Forest, and Linear Regression, we aim to forecast both the timing and type of future transactions. The primary objective of our model is to predict whether the next transaction will happen within 3 days, 14 days, or more than 14 days from the last transaction and also whether it will be an EFT outbank or a in-bank transfer.

The ability to accurately forecast these patterns has important implications for private banks. It can increase liquidity management, improve customer service through personalized offerings. Moreover, since data-driven decision-making is becoming crucial more and more, such predictive capabilities can provide a significant competitive advantage.

Finally, this study not only contributes to research on financial machine learning applications in banking but also offers practical insights that can be used by private banks to optimize their operations and better serve their customers.

Chapter 2:
METHODOLOGY

2.1 Data Collection and Preprocessing

The dataset used in this study consists of two years of electronic fund transfer (EFT) transaction data from a private bank in Turkey which begins in early 2023. The original dataset contained approximately 6 million fund transfers per day however we applied specific filtering criterias to focus on transactions with a high financial impact.

For corporate customers, we included only the transactions that exceeds 10,000 USD, and for individual customers only transactions exceeding 100,000 USD were considered. Aim for this is again is that in our analysis we want to focus on high-value transactions since they likely to have a more impact on the bank's operations and liquidity management.

The target variable for our first predictive model is 'days_until_next', representing the number of days until a customer's next transaction. This was calculated by determining the time difference between consecutive transactions for each customer. Also, a classification approach was tried to predict the ‘days_until_next’ target via discreatizing it into 4 categories such that (0,1], (1,3] , (3,14], and (14, 50) in order to represent spectrum of the 4 type of customers. Therefore, for both approach target variables are created. Figure 2.1.1 illustrates the distribution of continuous feature.

The target variable for our second predictive model is ‘next_trf_tp’ that represent the type of the next fund transfer, which is either ‘EFT’ or ‘Havale’. Distribution of ‘next_trf_tp’ can be seen in Figure 2.1.2

In order to ensure each customer have enough data we applied additional filters so only customers with more than 100 transactions in the two-year period were included which allows sufficient historical data per customer. Furthermore, we excluded customers with highly regular transaction patterns to focus on more variable behavior. Specifically, customers who have more than 40% of their transactions occurring at the same interval were removed because these regular patterns might skew our predictive models.

Data cleaning involved handling missing values, removing duplicates, and ensuring data consistency. Transaction dates were converted to a standardized datetime format for easier manipulation and feature extraction.

By applying these preprocessing steps we want to create a clean, balanced, and representative dataset focusing on high-impact transactions and customers with variable transaction patterns. This dataset is the main dataset which we do the feature selection and model development processes.

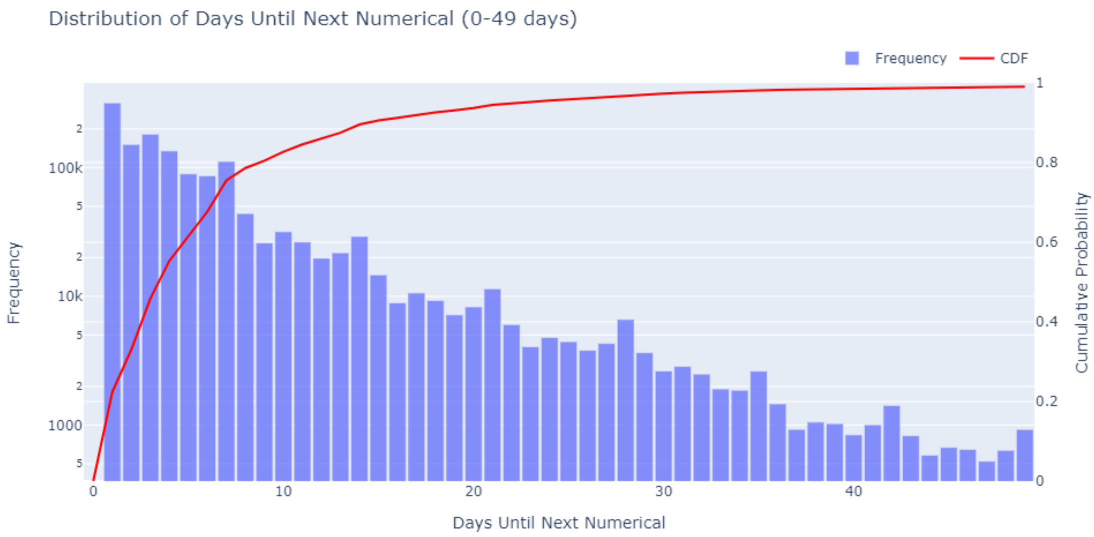

Figure 2.1.1: Distribution of continuous target variable

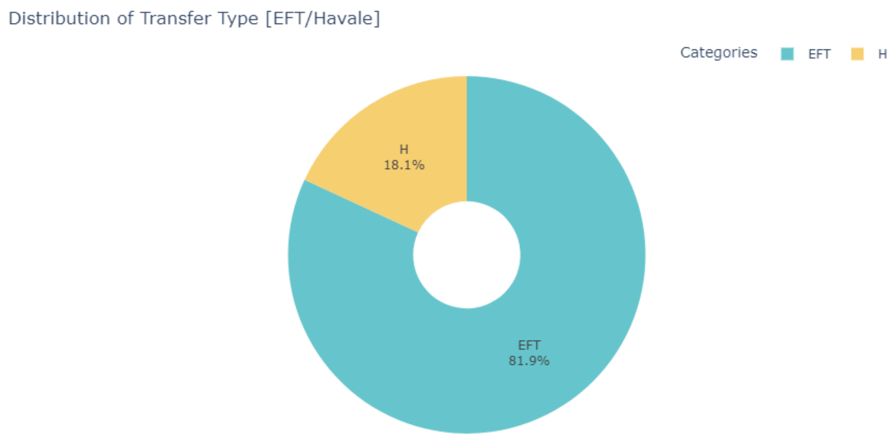

Figure 2.1.2: Distribution of categorical target variable

2.2 *Feature Engineering*

To improve the predictive power of our models, we created a set of features that representscustomer transaction behavior. For tree-based models, we created 'previous_days_between' features which record the intervals between the last several transactions for each customer so that the models can see recent transaction frequency patterns. We also added seasonal features such that day of the week, month, and quarter to include seasonal patterns in transaction behavior. Also we calculated cumulative features such as the total transaction count and total transaction amount up to each point in time. Additionally, we created rolling window features, including average transaction amounts and average days between transactions for various time windows (e.g., last 3, 7, and 30 transactions). These moving averages help capturing recent trends in customer behavior. These engineered features, with the original transaction data, form a feature set that aims to capture both long-term patterns and recent trends in customer transaction behavior, potentially improving the performance of our predictive models.

Train and test split was another important topic we focused on. We take the last three months of data, until 2024-04 as test data for each customer. A sample train-test split for a customer can be seen in Figure 3 below.

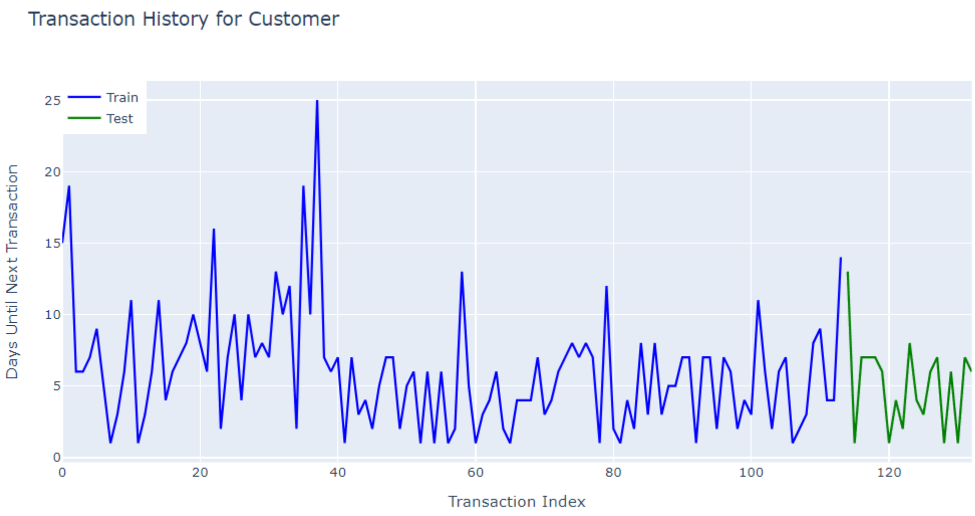

Figure 2.2.1: Train-test split for a customer

And in the subsequent data preprocessing step, continuous variables in both features and the target variable are standardized using a StandardScaler. On the other hand categorical variables in the feature set are handled via OneHotEncoder.

Chapter 3:
MODELS

In this reseach there are the two modelling problems which are first predicting the the days until next transaction and the second is that the type of the transaction which will happen. Therefore, we will analyze them in two distinct phase.

3.1 Days Until Next

To predict the days until next transaction both regression and classification approaches were tried. In the regression approach we predict directly to the how many days are until the next transaction. On the other hand, in the classification approach the target variable was discretized in to 4 categories of (0,1], (1,3] , (3,14], and (14, 50) in order to represent 4 type of customers.

3.1.1 Regression Approach

In the regression approach 3 and 1 naïve model was tried to predict the days until next transaction target variable. Naïve model is taken as a model predicting a three transaction rolling mean to benchmark other models. The other three models were XGBoost, RandomForest, and an LSTM Network. Both XGBoost and RandomForest models were trained using a grid serach cross-validation technique to tune the model’s hyperparameters. The hyperparameters tuned during the cross-validation for the models are shown in Table 3.1.

For the LSTM model, for each customer a sequence of length 10 was created consisting of autoregressive and engineered features. This was fed into an LSTM network in batches with EarlyStopping and ReduceLROnPlateau. Dropout layers are also used to prevent overfitting. All three models are subjected to feature selection, hyperparameter tuning. In the LSTM model, network architecture is tuned to perform the best metrics.

Table 3.1: Optimization Hyperparameters for XGBoost and RandomForest Models

Model	Hyperparameters
XGBoost	'max_depth': [3, 5, 7], 'learning_rate': [0.01, 0.1, 0.3], 'n_estimators': [100, 200, 300], 'min_child_weight': [1, 3, 5], 'subsample': [0.8, 1.0], 'colsample_bytree': [0.8, 1.0]
RandomForest	'n_estimators': [100, 200,], 'max_depth': [None, 10, 20], 'min_samples_split': [2, 5], 'min_samples_leaf': [2, 4], 'max_features': ['auto', 'sqrt']

3.1.2 Classification Approach

In the classification approach LogisticRegression, XGBoost and SVC are applied to the categorical target variable. Hyperparameter tuning, feature selection are tried to achieve best performed models. For the benchmark model a VotingClassifier of the last three categorical target is applied.

3.2 *Transaction Type*

For the second model where we want to predict the next Transaction Type multiple models and a naïve model are used. In the naïve approach a VotingClassifier of the last three transaction type is applied and this model is used as a benchmark to other models. The main models we tried to predict the Transaction Type were LogisticRegression, XGBoostClassifier and SVC. Since the data was higly imbalanced, several approaches are tried to over come the imbalance such as class_weight, and SMOTE. All models are subjected to feature selection and hyperparameter tuning.

Chapter 4:

RESULTS

4.1 Days Until Next

The results obtained from k-fold crossvalidation for both regression and classification approach were higher than the benchmark model. In the regression approach, we obtained 6.54 RMSE in XGBoost Model with a R2 of 0.1688. Similarly, in the both RandomForest and LSTM models RMSE's were 6.51 and 6.59 with 0.1745 and 0.1441 R2's respectively.

Table 4.1: Metrics for regression model.

	Rolling Mean 3	XGBoost	RandomForest	LSTM
MAE	5.08	4.05	4.09	**4.016**
RMSE	8.99	6.54	**6.51**	6.59
R2	-0.57	0.16	**0.17**	0.14

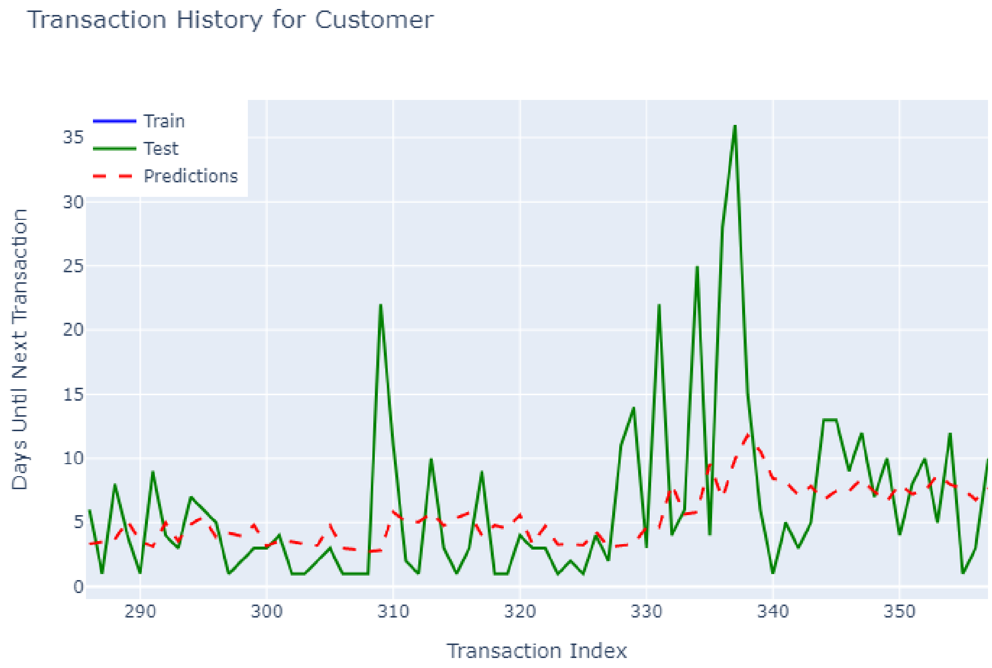

Figure 4.1: Predictions made for a sample customer using RandomForest

For the classificaiton approach, we compared accuracy and f1-scores for the 4 categories. While in the VotingClassifier we get 0.35 accuracy, for the LogisticRegression, XGBoost and SVC we get 0.45, 0.49, and 0.46 respectively.

Table 4.2: Classification report for XGBoost.

	precision	recall	f1-score	support
0	0.47	0.45	0.46	65901
1	0.45	0.28	0.34	68819
2	0.52	0.75	0.61	120580
3	0.40	0.02	0.04	27701
accuracy			0.49	283001
macro avg	0.46	0.37	0.36	283001
weighted avg	0.48	0.49	0.45	283001

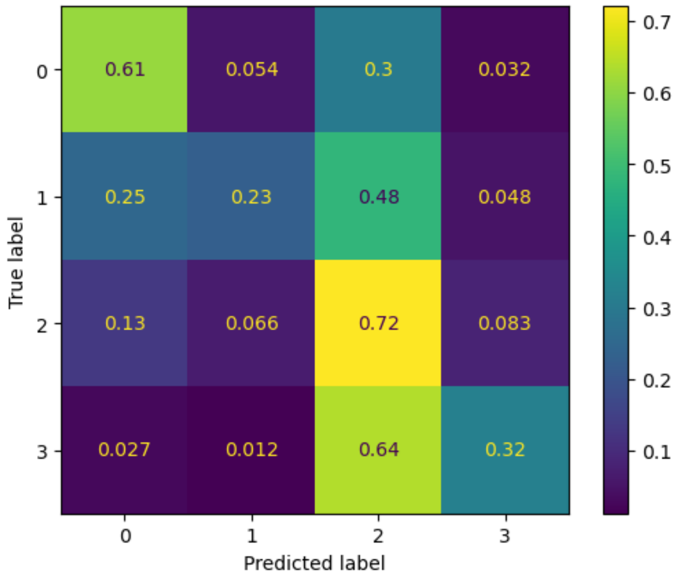

Figure 4.2: Confusion matrix for XGBoost model

An examination of the confusion matrices revealed that models performes better than naïve predictions however class 3 were consistently misclassified which suggest a level of difficulty of predicting transaction interval of (14,49]. This indicate that there is need for more features to distcriminate the customers who made transaction in longer periods.

For the second classification approach we choose our target as binary such that whether the days until next is within 14 days or higher in order to differentiate the 'semi-churning' customers. Transaction Type

Table 4.3: Classification report for XGBoost.

Classification Report:

	precision	recall	f1-score	support
0	0.98	0.62	0.76	236519
1	0.28	0.90	0.43	38477
accuracy			0.66	274996
macro avg	0.63	0.76	0.59	274996
weighted avg	0.88	0.66	0.71	274996

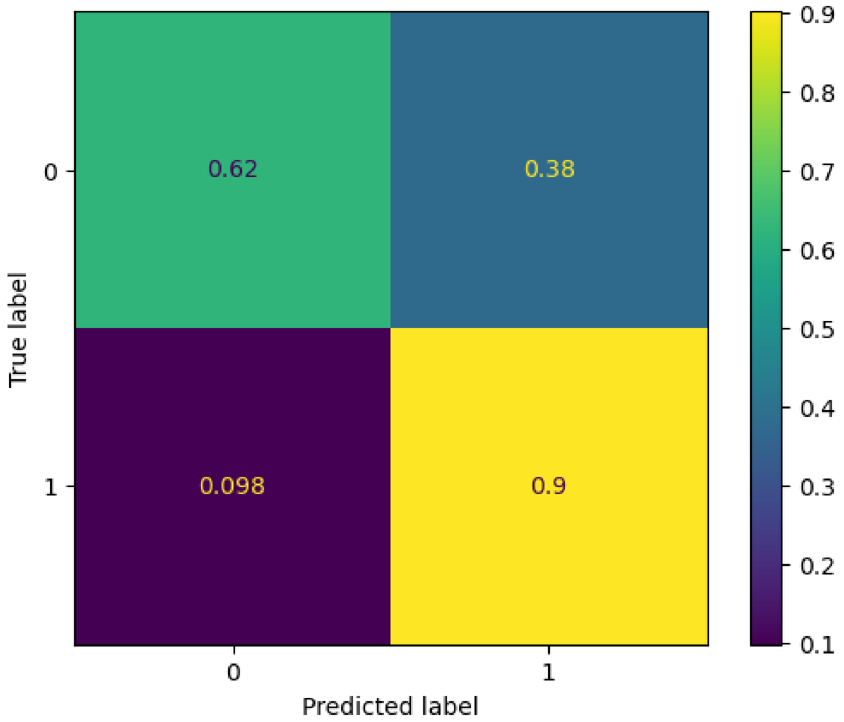

Figure 4.3: Confusion matrix for XGBoost model

In the second model which we predict the next Transaction Type we have an imbalanced classification problem with two classes of weights 0.80 and 0.20. Again the results are obtained from a k-fold cross-validation and three models have been tried to achieve best performing model for classifying whether the next transaction will be an in-bank or out-bank transaction.

For the benchmark a naive classification of VotingClassifier of the last three transaction type has been taken. Precision and recall of the in-bank transfers were 0.57 and 0.47 respectively.

In the Logistic Regression model we see a worst model than naïve model with a 0.43 precision and 0.07 recall with an AUC score of 0.62. In the logistic regression model we tried hyper parameter tuning also used class_weight to solve the imbalance problem. In the XGBoost the model performed worst than logistic regression with both SMOTE, RandomUndersampling. Metrics were 0.30 precision, 0.02 recall and an AUC score of 0.57. ROC curve for Logistin Regression can be found in Figure 4.3.

We can conclude that it is very likely the Transaction Type cannot be predicted from the features we have currently.

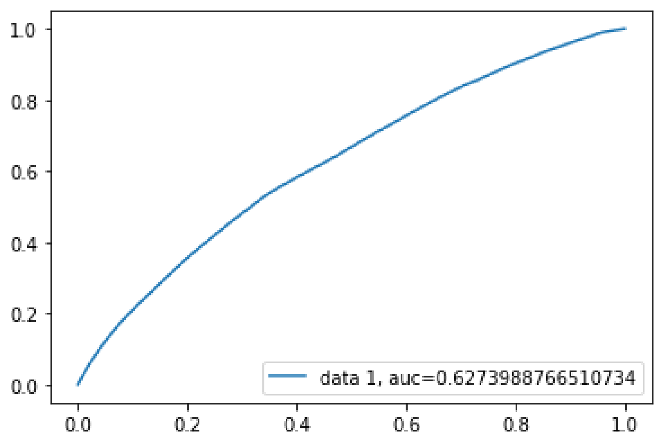

Figure 4.3: ROC curve for Logistic Regression model

Chapter 5:
CONCLUSION

In this study we explored the application of machine learning techniques to predict electronic fund transfer (EFT) patterns in private banking and we focused on forecasting both the timing and type of future high-value transactions. The research used several models such as XGBoost, Random Forest, Logistic Regression, and LSTM networks to predict two main target variable which are the number of days until a customer's next transaction and whether the next transaction would be an in-bank or out-bank transfer. A dataset of two years of EFT transaction data from a private bank in Turkey is used and specific filtering criteria is applied to focus on high-impact transactions.

To predict the days until the next transaction, both regression and classification approaches were tried. In the regression approach the best performing models (XGBoost, Random Forest, and LSTM) achieved RMSE scores between 6.51 and 6.59 where they outperform the naive benchmark model. The classification approach which categorized transaction timing into four classes achieved accuracies between 45% and 49% for the best models. They also outperform the benchmark. However, all models struggled to accurately predict transactions occurring after 14 days. This suggests that a need for additional features to better differentiate customers with longer transaction intervals. In that case we tried to catch the ‘semi-churning’ part of the customers where the customers make regular transactions however sometimes they tend to make transfers longer than 14 days periods. In order to do that we create a model with binary target variable which improved the performance in where the first model fails to predict.

To predict the type of the next transaction (in-bank vs. out-bank) the models faced challenges due to class imbalance problem. Although there has been attempts to solve this issue using techniques like SMOTE and class weighting, the best models only marginally outperformed naive prediction methods. The logistic regression model achieved the highest performance with an AUC score of 0.62 but the results show that predicting transaction type may require additional or different features than those currently available. This research provides insights into the potential and limitations of using machine learning for EFT pattern prediction in private banking which then highlights the areas for future investigation and improvement.

BIBLIOGRAPHY

Central Bank of the Republic of Turkey. (n.d.). Electronic Fund Transfer System and Electronic Securities Transfer System. Retrieved August 13, 2024, from https://www.tcmb.gov.tr/wps/wcm/connect/EN/TCMB+EN/Main+Menu/Core+Functions/Payment+Systems/Payment+Systems+in+Turkey/Electronic+Fund+Transfer+System+and+Electronic+Securities+Transfer+System

Keras team. (n.d.). Timeseries forecasting for weather prediction. Keras. Retrieved August 13, 2024, from https://keras.io/examples/timeseries/timeseries_traffic_forecasting/

Tang, Y., Song, Z., Zhu, Y., Yuan, H., Hou, M., Ji, J., Tang, C., & Li, J. (2023). A survey on machine learning models for financial time series forecasting. Journal of Neurocomputing, 512, 363-380. https://doi.org/10.1016/j.neucom.2022.09.003.

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). Dive into Deep Learning. Cambridge University Press.