A Guidance Framework for Resolving Problems of Credit Card Churn Detection Systems

Fehim Altınışık Ahmet Sayar

Abstract

The study examines possible solutions to resolve problems that occur in the process of operationalizing systems that predict the financial services that customers will terminate using machine learning and business intelligence approaches. The scope of the study consists of defining the problem, collecting and integrating the data, training the models, evaluating and validating the outputs, and making them ready for use in the production environment. In addition, intuition about infrastructure, architecture, processes, technologies, and other artifacts used during the study is included. The data manipulation and pre-processing framework proposed in this study is applicable to both real and synthetic banking data. To implement each step in detail, an improved version of an auxiliary study was used. A study has been carried out in a financial institution in Turkey, chosen as an auxiliary, in which customers who are likely to cancel credit cards are determined by machine learning. The problems, findings, and results are examined in detail. The framework used in this study is believed to be used not only in the integration of credit card product churn detection systems but also in the integration of other systems that use machine learning and deep learning.

Keywords: Machine learning, artificial intelligence in operation, credit card, churn detection, data synthesis.

1 Introduction

Turing's work [1] made it possible for "machine learning," also called "artificial intelligence," to be used to solve problems in everyday life

doi:10.56415/csjm.v33.02

instead of using clearly defined or programmed expressions [2]. We are faced with the problem of constant or periodic change, renewal, or updating of the data, which affects the results and performance of the predictions and classifications that we carry out with machine learning and deep learning models. The models built using machine learning make unrealistic predictions and classifications as the environmental conditions and inputs in the systems change, although they achieve high success rates in a short period of time. At the end of the short time intervals, data scientists iterate through the same process of collecting, organizing, and integrating data to increase the success of the models again [3]. As a result, acquisitions in the field of machine learning are once again labor-intensive and inefficient.

However, when attempting to integrate most of these intelligent data-driven systems into software engineering projects, problems arise because the validity of the data expires, conflicts arise, or the integration does not meet the needs of the time. Software automation is needed to control the reconstruction, organization, and integration. The tasks of such software are as follows: (i) integration and updating artifacts; and (ii) maintaining the models.

To demonstrate the impact of such software automation, we have chosen a credit card churn prediction problem as an auxiliary use case. When we examine the credit card churn detection problem, changing economic indicators over the years and the amount of credit card customers' expenditures on education, health, and seasonal changes cause the data of the customers to change over time. Changes in the characteristics of credit card customers' data bring about the problems mentioned above and make it difficult to carry out data mining studies or reuse the models trained with the same data. To build a maintainable and sustainable solution for the credit card product churn prediction task, these problems should be eliminated by transferring model management to the machine learning lifecycle components. The previous work [4] is the cornerstone of this study. The stages of collection, organization, integration, training, data testing, model evaluation, and deployment of models in production environments were examined, respectively. The architecture to be examined throughout this study is believed to be used not only for periodic classification problems but

also to maintain other classification and prediction tasks. Given the scarcity of open datasets in financial services, we propose a straightforward approach to synthesizing financial datasets. Section 2 expands on the definitions of the aforementioned problems.

The proposed software automation introduces a paradigm shift by integrating the machine learning lifecycle into model training and prediction steps, contrary to the standalone utilization of models like Gradient Boosting Machines (GBM) and Random Forests. Although GBM and Random Forests' standalone utilization can handle complex datasets and achieve high predictive accuracy, they often lack mechanisms for continuous adaptation and refinement in response to evolving data and environmental changes. By automating tasks such as data reconstruction, organization, and integration, the proposed approach not only streamlines model maintenance but also enhances adaptability and scalability. This strategic integration of the machine learning lifecycle addresses persistent challenges related to data validity, conflicts during integration, and model maintenance over time, ensuring sustained model effectiveness and relevance in real-world applications.

In this study, to define the foundation of the machine learning life cycle, the open-source ml-flow library was utilized. It tracks experiments, facilitates collaborative code usage, and provides unified control over various ML libraries. The project architectures vary, but the implemented architecture separates the experimental work and the model packaging, as depicted in Fig. 1, with ML flow connecting model performance logs and metrics between these phases.

We organize the rest of this article as follows. The definition of the problem is given in Section 2. Related work on the subject is presented in Section 3. The processes for data collection, processing, and integration are mentioned in Section 4. Section 5 presents the details of the models used. Section 6 describes the integration and architecture of the system, and Section 7 concludes the paper.

2 Definition of Problem

2.1 Detection of Customers Tending to Cancel Credit Cards by Machine Learning

According to data from the Interbank Card Center, the total number of credit cards in Turkey exceeded 68 million in Q3 2019. Competition among institutions intensifies as digital channels accelerate customer acquisition, leading to increased service cancellations and churn. Reichheld [5] suggests implementing defensive policies to retain satisfied customers rather than targeting low-value ones. It is crucial to promptly address negative experiences, given the swift cancelation of credit cards post-negative encounters. This study aims to predict credit card churn, achieving positive results. Customers conforming to this definition were termed "churners", while those continuing card usage were termed "survivors". Definitions of Nie et al. [6] classify evaluation periods into short (3 months or shorter), medium (3-6 months), and long-term (6 months or longer). However, unlike Nie et al. [6], performance periods are based on these intervals, not on a year. The observation period extends until the model is used, while the performance period denotes the waiting period to evaluate the prediction results after executing short-, medium-, and long-term predictions regarding the credit card customer's status.

2.2 System Integration and Synthesizing a New Dataset

Before the implementation of the system in production environments, models must undergo testing, validation, and approval by model owners. They should accurately classify samples in a benchmarking dataset, with only those performing above a defined threshold chosen for deployment. The models' performance and other evaluation criteria are stored in databases to support decisions made by components of the automated machine learning life cycle.

Availability of datasets that most accurately represent the customer may not always be possible in the financial services domain, since authorities around the world strictly regulate the distribution and donation, even of anonymized financial data. An alternative for researchers

who have difficulty accessing real banking data is to combine Practice on Knowledge Discovery in Databases 1999 [7] and World Bank data. A brief way to use these datasets or any other datasets available to researchers is introduced in Section 4.

3 Related Works

The interest in studies to classify customers who tend to terminate the use of credit cards or other financial services has intensified in the last two decades. Poel and Lariviere [8] published their findings showing that the reasons for customer churn can be determined using multiple proportional risk models. In their work, independent models were built using demographic, macroenvironment, perception, and behavior data collected from the leading banks of Europe. The effects of these factors on the relationships of customers with the institutions they work with were analyzed separately. The results of the study claim that the tendencies of customers to churn depend more on environmental factors than their spending routines with credit cards. This finding explains the misclassified examples with high cosine similarities discussed in Section 5. Another point that is noteworthy in the study is the difficulty of collecting perceptual – paid reasonable, general customer experience – data on an individual basis. Zopounidis [9] studied risk and probability models such as Mavri and Ioannou [8], Poel and Lariviere [8], and the causes of the churn problem. Since the target audience selected in this study represents the trends of citizens of a single European country, the results obtained are not included in the data research stages of this article.

Lemmens and Croux [10] have shown that 'boosting classifier trees' can distinguish scattered target classes by applying an appropriate bias correction. To evaluate the outputs of this algorithm more efficiently, a case study similar to the one applied in the telecommunications industry should be applied to the financial services industry. Another aspect that distinguishes this study from the others is that it claims that the target classes selected for the training data should be sampled in equal quantities relative to each other.

The performance of the logistic regression and decision tree algo-

rithms was examined in detail in the studies by Nie et al. [6], and it was reported that these algorithms were successful in solving classification problems when the classes in the datasets were approximately equal. The high classification performance of these algorithms when the data sets formed evenly was also mentioned in the previous study. The rates used to train the models in the study are 1: 1, 1: 2, 1: 4 and are similar to the numbers discussed in Section IV. However, as previously stated in the practical application, the ratio of target classes to one another is around 1: 36 and it is far from the numbers that this study deals with. The study also contains details that will form the basis for data sets to be prepared to solve customer churn prediction problems.

In the approach proposed by Lin et al. [11], the rule bases drawn from the cluster theories have been operated on network maps, and successful results have been obtained on an equal number of randomly distributed target samples. However, the outputs of this method have been integrated into the institution's CRM systems but have not achieved the expected success in deployment. In the study of Chen et al. [12], instead of working with transactional data from customers, as in their previous work [13], behaviors were displayed as a state or a target for another state (for example, survival to churn). It is claimed that predictions can be made by examining the traces.

He et al. [14] also addressed the negative effects of the fact that the data sets were not evenly distributed across the target classes due to the nature of the problem. In this study, support vector machine classifiers have been used. Linear, RBF, and Sigmoid functions, which are the most frequently used kernel functions, have been used when building the model. However, the scarcity of customers belonging to the "churner class" has negatively affected the success of the model. According to Xiao et al. [15], the transfer learning methodology can solve the customer churn problem. The use of transfer learning methodology to solve the problem in the study was also found to increase the success in imbalanced data sets. However, it was also stated that the data sets used in the study lack the required features specific to target sectors (finance, health, telecommunications), such as credit ratings, information on special customer expenditures, examination results, or other information in the telecommunications sector, and this situation

negatively affects the overall performance of the model. The dynamic feature selection algorithms used in the study were tested and it was stated that the algorithm performed well.

In addition to the previous work of Rajamohamed and Manokaran [16], the rough K-Means algorithm was used with the Support Vector Machine, Naive Bayes, Decision Tree, and the Random Forest Classifier also used in the article, with a data set consisting of 23 features, which gave better performance. In the data set used, the ratio of the target classes to each other is around 1:4 and therefore, does not meet the requirements regarding class imbalances.

The above studies provide valuable feedback on the stages of forming data dictionaries and training classifiers. However, the data used in most studies consist of samples provided by financial institutions in foreign countries or by [7]. For this reason, there is not enough information about the pre-processing and aggregation stages of the data. For customers using two or more credit cards, the steps discussed in Nie et al. [6] were followed when aggregating transactions on credit cards. Tables 1 and 2 show examples of aggregation operations.

Table 1. View of customers using two or more credit cards

CUST_ID	CREDIT_CARD_ID	BALANCE	LAST_USE
#100	#150	1500	12.07.2018
#100	#160	1200	13.09.2018
#200	#250	300	21.11.2017
#200	#260	2100	11.05.2019

Table 2. Aggregated view of customers using two or more credit cards

CUST_ID	MIN_BALANCE	BALANCE	LAST_USE
#100	1200	2700	13.09.2018
#200	300	2400	11.05.2019

4 Data Manipulations

To achieve the required level of performance in the prediction problem, the models must be trained with recent and historical data that represent the profiles of the institution's customers. The data in this study were organized in a tabular format that represented the summary of 12 month credit card transactions, backward from a predetermined date.

In this study, the fields describing the amount of payment in Yeh, IC [7] and the fields describing the transactions in PKDD'99 were selected as similar and used to merge these data sets together. To merge, common or similar fields must initially be normalized. After normalizing, records with the highest similarity were outer joined around similar fields. Finally, other fields are combined around similar fields. Researchers could apply other possible validation methods to improve the validity of this operation. Even if it is possible to generate a high number of transaction-based features using only PKDD'99, the motivation of the above operations was to demonstrate a way to include demographic features from another dataset. Using transaction-based features generated using PKDD'99 and World Bank data, interest rates specific to each customer could be generated. After these operations, every combined record could be used as data for a single customer. Even if data sets lack features related to customer experience and loyalty using only product-based, transaction-based, and demographic fields in open datasets, 80 of 143 features used in this work, which were obtained from real banking data, could be derived. The results of experiments carried out with these synthetic data are shared in Section 5. Researchers are recommended to study the synthetic credit card transaction generation methodology proposed by Altman [17]. Such data synthesis methodologies could also support different studies in the field like Pehlivanlı et al. [18].

Durations between transactions in PKDD'99 are adjusted due to contemporary credit card usage patterns. Not all preprocessing steps described below may be necessary for a synthetic dataset, which is typically more structured than modern financial data. Although tree models used here generally do not require normalization, they should be applied during testing with synthetic data due to their different natures.

4.1 Target Dataset Selection

Instead of training a single model for institutions with diverse customer behaviors, segmentation is performed and separate models are run for each group. This study will employ a single model after filtering eligible customers. Customers without stable income, seasonal agricultural workers, limited credit card users, and similar cases are excluded from the classification. Following Nie et al. [6], credit card actions for 6 months or more are removed from the dataset, updating the parameter from one year.

The definition of churner, which we have made in Section 2, needs to be expanded here for easier examination of extreme cases. To include a sample (customer) in the survivor class, it should be a necessary and sufficient condition that the customer has made transactions with at least one credit card in the last 12 months. According to this definition, if a customer has at least one credit card that has been used in the last 12 months, he will be included in the survivor class even if he has canceled one or more credit cards. Through the use of stratified K-fold cross-validation to preserve the natural class proportions, any potential biases are minimized, thus establishing a robust basis for further analyses.

The study also prioritizes the utilization of anonymized public financial data, complies effectively with the GDPR regulations associated with bank data, and guarantees the compliance with privacy standards.

4.2 Feature Exploration

The data sets examined in [7] and other works contain demographic, transaction-based, customer experience, and information about the client's product portfolio. In their analysis, Poel and Lariviere [8] presented the hypothesis that demographic, transaction-based, and perceptual data did not have a direct impact on the customer's churn. Using techniques in the section and data set of this study, it was observed that demographic data did not correlate with the trend of churn but showed high parallels with perceptual data, in line with the hypotheses of [8]. Customer experience data are derived from information about how customers use the organization's communication channels, how

frequently they use them, and whether their requests are met.

Before the study, each record had 535 feature sets that explained the perceptual, transactional, and portfolio data of the customers. This data summarizes the customers' relationships with the institution. The variance ratio and the Gini impurity have been used to determine the features that would be useful for the success of the model. Of the 535 features collected and derived, those that would not positively affect the performance of the model were removed from the set. To determine the features to be eliminated from the data set, Pearson's matrix, Correlation matrix, Univariate analysis, and linearity analysis were performed. After eliminations, 204 features remained. Gini Importance Analysis and Variance Rate Analysis were used to find features that were of higher importance than the remaining 204 features. These analyses aid in identifying the most influential features and guide subsequent modeling efforts.

The Gini index was used as an impurity function, as described by Louppe et al. (2013). When used as an impurity function, the Gini impurity determines the features to be used in nodes in decision trees by sorting the features according to their significance scores that add up to 1.0. Following the examination of the features, 143 of the remaining 204 features decided to be included in the data.

The variance ratios of the remaining 204 features were examined to verify the results of the Gini Importance Analysis using the principal component analysis proposed by Tipping & Bishop [19]. The results of the variance analysis show that 45 of the features can explain around 95% of data. However, it has been decided to use all 143 features obtained using Gini Impurity. When only these 45 features are used in the model, the accuracy values obtained are high, while the F-score values and model lifts are low. Customers in the churner class, which have a low presence in the classification problem, form a small portion of the data set but are the building blocks of the problem themselves. The increase in the number of false positives caused by the misclassification of these customers does not greatly affect the baseline accuracy values, but it decreases the F score value. We use the remaining 98 features despite their only explaining 5% of the data because they help classify members of the churner class with more precision.

4.3 Sampling and Structuring Data

This step sets the stage for effective model evaluation. Data sets are formed using data from churned and surviving customer groups. Each customer's performance period spans the observed month and the following 11 consecutive months. This period aligns with the target period, characterized by a low variance in customer data due to economic indicators. All customer performance periods within the dataset fall within this target period (see Table 3.). In accordance with the way in which customer groups are formed (segmentation) or other parameters, the duration of the observation period can be defined as any of the periods specified in Section 2.1. In this study, the duration of the target period was determined to be 24 months. The interval of the target period variable needs to be updated in the long term, considering economic indicators and customer behavior.

Table 3. Performance, observation and target periods

Sample	Target	Performance	Observation
#110	01.04.2017–01.04.2019	15.04.2017–15.04.2018	6-9 months
#120	01.04.2017–01.04.2019	15.07.2018–15.07.2019	6-9 months
#210	01.04.2017–01.04.2019	15.01.2018–15.01.2019	3-6 months
#230	01.04.2017–01.04.2019	15.03.2018–15.03.2019	1-3 months

Data sets to be used for training were formed by combining churner and survivor customer data samples in ratios of 1/1, 1/2, 1/4, 1/5, 1/9, 1/16, 1/25, and 1/50. Success metrics were observed for models trained with each combination. The most effective ones for the training dataset were observed to be those combined in ratios of 1/1, 1/5, and 1/9. According to corporate policies, one of the coverage rates or model lift formulas can be used as success criteria. In this study, a model lift was used as a success criterion, and formula (1) was used to define model lift L, where the Natural Ratio is the percentage of the target class in all classes. Table 4 shows the impact of sampling on the success of the model.

$$L = \frac{Precision\ Ratio}{Natural\ Ratio} \tag{1}$$

Table 4. Impact of samplings to the model success

Samplig ratio	Precision ratio	Recall ratio	Model lift (L)
1:1	16.85%	58.79%	6.02
1:5	29.72%	39.82%	10.61
1:9	33,88%	32.63%	12.1

4.4 Data Preprocessing

In this work, the data preprocessing steps consist of outlier defection, imputation, and SMOTE operations, respectively. These preprocessing steps contribute to improved model robustness and generalizability.

To prevent the imputation operations performed using the average or most repetitive values from being affected by the outliers, the outliers were first removed from the dataset. This is the same definition used by Bluman [20] for the term outlier. The reason for the high number of outliers in the customer data set is correlated with the high number of features used or the lack of data assurance practices required when aggregating data. If any attribute in the data set contains an outlier value, the attribute is replaced on a rule basis or by values determined by expert opinion. The rarity of the samples in the churner class is the main reason for using this method of outlier defection instead of eliminating the sample.

To fill in features that cannot be derived due to loss or dependence on other features, rule bases, expert opinions, and the robust regression technique proposed by Rousseeuw and Leroy [5] are used. The rarity of examples in the churner class is a valid reason to use resource-consuming algorithms, such as robust regression, instead of eliminating the sample. When performing imputation operations on lost fields with these methods, the order of imputation also must be followed (see Table 5).

SMOTE, proposed by Chawla et al. [21], was applied to data sets after sampling and structuring them. The data set has been resampled to favor customers leaving, and the class ratio has been reduced to 1/1. After the SMOTE process, the lifts of the models trained with the 1/9

Table 5. Example of pseudo respective feature generation

Feature	F2	F3	F4
Formula	F1 * F7 / F4	F2 + F4	F7 / F8

dataset, where the best sampling was done with model lift, increased to 12.1. The samples in the survivor class have not been grouped, and a single model is used to classify all of them. Thus, more samples belonging to the survivor class existed. The SMOTE operation must be applied to the synthetic data set to make its class proportions closer to those of the real financial data set.

4.5 Feature Dictionary

Only 10 of the significant features used in accordance with the data privacy policies of the institution where this study was conducted are shown in Table 6.

Table 6. Sample of features and descriptions

Feature	Description
X112	The amount of interest offered for card statement
X81	Number of direct debit orders
X32	Credit card age (months)
X24	Number of communications established with call centers in the last year
X12	Credit card debt amount
X74	Average return time of the call center to the customer
X178	Number of branch visits in the last 3 months
X54	Duration of the longest call with the call center
X43	Total number of purchases made with credit cards in the last 3 months
X98	Total amount paid on purchases made with credit cards in the last 3 months

5 Tree Learning Models

This study uses random forest and gradient boosting algorithms, prevalent in banking and finance. Taking Type 1 error as a false positive and Type 2 error as a false negative, other formulas will be computed

to compare the success criteria. Accuracy and baseline accuracy values pose challenges in model interpretation as a result of the continuous dominance of the "survivor" class over the "churner" class. Although 90% of the surviving customers can be accurately classified, maintaining low false-positive rates for the churners is difficult. Hence, precision and recall formulas, excluding correctly classified surviving customers, are utilized as metrics when assessing model success via the confusion matrix.

$$Precision = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN}. \tag{2}$$

The models produce scores for each of the churn and survivor customer classes that range between 0 and 1, where the scores add up to 1.0. The decreasing trend observed in the number of churner class as the confidence threshold increases is included in Table 9. The confidence threshold value in Table 9 is the lowest score that the classifier must give in favor of the churner class. In this way, the customer can be classified as a churner with more precision. Another point that should be taken into consideration while examining Table 7 is that instead of using a training data set with a class ratio of 1/9, a data set with a natural ratio of 1/36 is used as the test set. In addition, since estimates are made with customers for 2 or more periods in the same target period in the test sets, the natural rate is decreased to 0.623%. All models in this and Conclusion sections were first trained with datasets that were sampled in a 1:9 ratio of the classes, and then reproduced in favor of the non-dominant class, by using the SMOTE method and resampled in a 1:1 ratio.

Table 7. Confusion matrix for synthetic financial data

		Predicted			
		Random Forest Classifier		Gradient Boosting Classifier	
		Survivors	Churners	Survivors	Churners
Actual	Survived	29806	37	29790	44
	Churned	4	153	20	146

Table 8. Confusion matrix for real financial data

		Predicted			
		Random Forest Classifier		Gradient Boosting Classifier	
		Survivors	Churners	Survivors	Churners
Actual	Survived	28620	1192	28078	1734
	Churned	45	143	96	92

5.1 Gradient Boosting

The gradient boosting algorithm attempts to optimize low-success models in a group of two or more models using different or identical learning algorithms [22]. Gradient boosting performs better than random forest when evaluated with the test dataset. It is observed that the models trained with the random forest algorithm initially do not reach the accuracy threshold required by the gradient boosting algorithm. However, after deployment and long-term testing, it is documented that the gradient boosting algorithm memorizes the training data.

5.2 Random Forest

Random Forest is an extended version of the bagging classifier approach [23]. Since the ID3 algorithm does not allow the use of continuous numerical data, the C5.0 and CART algorithms were used in this study. When using C5.0 and CART trees, there is no need to convert continuous numerical data into categorical data. When evaluated with a test set, the models trained with the Random Forest algorithm showed less performance than the models trained with the gradient boosting algorithm. However, it is observed that they have achieved higher performance in the evaluation made with periodic and verification data.

Table 9 shows that the confidence threshold should be kept at 0.80 and above to achieve the highest accuracy and model lift in the classifications made with models trained with the random forest algorithm. Table 10 shows the number of customers that are correctly classified as churners and the distribution of these numbers by month. According to the results obtained, the gradient boosting models predict churn-

ers with higher accuracy scores in the short term for the institution. However, in the long run, the random forest model performs better. About 70% of the customers classified by models trained by the gradient boosting algorithm churn within the first 3 months after prediction. In models trained with the Random Forest algorithm, this number remains around 50%.

Table 9. Effect of confidence level over other performance metrics

Confidence Levels	Predicted	Actual	Precision	Recall	Lift
0.80+	341	87	%25.9	%46.9	9.25
0.60+	874	128	%14.6	%68.1	5.21
0.50+	1336	143	%10.7	%76.3	3.82
0.40+	2024	156	%7.8	%83.4	2.79
0.20+	4997	175	%3.5	%93.6	5.6

Table 10. Churn periods of customers after classification

Model	Number of Total Churners	Churn Period		
		In 3 months	In between 3 - 6 months	After 6 months or later
Gradient Boost	92	64	19	9
Random Forest	143	79	39	25

The models trained in this study were tested using separate test sets sampled from real and synthetic banking data, rather than a common validation set. When Tables 7 and 8 are examined, it is seen that the classes in both data sets are classified more successfully using Random Forest Classifiers. This development can be considered as an indicator of consistency. However, false-positive numbers in the results obtained from synthetic datasets are too low to be obtained in a real test environment. None of the companies that set the industry standards today can guarantee that they can achieve such a low false-positive rate. To understand the reason for this, the most determinant features of the model trained with the synthetic data set were compared with the 30 most determinant features in the real banking data. Among the most decisive features in real banking data are the features that measure customer satisfaction and customer experience.

The synthesized data set does not contain the features that measure customer satisfaction and customer experience, as mentioned before. For this reason, a model trained with synthetic data can predict the cancelation of credit card transactions, but cannot predict satisfaction and experience cancelation under real-world test conditions.

5.3 Evaluation

The stratified K-fold algorithm is an algorithm in which the natural proportions of the classes are maintained during the formation of folds from training and test datasets [24]. In this study, stratified K-folds were used to eliminate the deviation effect before training the models. The maximum difference between the accuracy of the models and the precision rates is determined to be 2.7%. In addition to the accuracy rate, the precision rate, and the lift of the model, in this study, a further criterion was used to compare the success of the models. The samples that were misclassified as false positives or false negatives using models trained with the algorithms in this study and other algorithms used in the previous work were compared with each other with a cosine similarity kernel [25] that gave valuable insight about possible reasons for misclassification. If the customers trained by the Random Forest algorithm are classified with a confidence threshold of 0.50, the number of customers classified as false positive is $M_1 = 1192$, and the number of customers classified as false negative is $M_2 = 45$. $M_1 * M_2$ similarity ratios were obtained with the prediction results of the members of these two groups. More than 91% of these similarity ratios were on a scale of 0.8 or higher. The results suggest a requirement for models with enhanced feature sets, enabling a deeper exploration of the samples to effectively discern between classes.

6 Deployment and Model Life Cycle

6.1 Data Centered, Data Flow, and Hybrid Architectures

The application must meet the requirements of Sections 1 and 2, demanding architectures suited for data movement and processing.

Adopting a hybrid approach by merging Data-Centered and Data-Flow architectures enables diverse model functionalities and monitoring.

The Data Centered Architecture centralizes data storage for frequent access by peripheral units. It ensures data consistency by allowing modules to interact through stored data modifications. The system comprises central data and data accessors, where customer data mirrors central data, and machine learning models reflect data accessors. The architecture chosen adopts features from the Warehouse model, managing data accessor components via a web service application.

The Data Flow Architecture processes data sequentially and independently, copying, modifying, and storing it as needed. Communication between components is facilitated by graphs or flow diagrams. This architecture is ideal for operations on structured data, and the application inherits some of its features, including batch sequence preprocessing.

The hybrid architecture, which inherits traits from two distinct approaches, was outlined on the basis of the aforementioned roles. Processes related to data integration and model operation were addressed separately. The resulting system allows customized data processing stages and integration with any required machine learning algorithm, fostering flexibility and compatibility with other modern AI model management services like ML-flow.

The application utilizes multiple models to generate meaningful outcomes, each with its own controller objects, which facilitate interaction with various learning algorithms. These controllers manage data preprocessing, model training, and result recording systematically. Users determine the deployment of the model, and controllers function as data accessors in a data-centric architecture. ML-flow REST API[1] was implemented to interact with controller objects or provide interfaces for users.

6.2 Technology Stack

Operational efficiency is a key aspect of production systems. Thus, incorporating program control structures into native database operations

[1] https://www.mlflow.org/docs/latest/index.html

has proven to be a crucial challenge. In selecting storage and database technologies, support for writing Stored Procedure and Stored Function objects is crucial for incorporating program control structures into SQL statements, enhancing system flexibility. In addition, optimizing data preprocessing operations in the production environment can significantly benefit system analysts and architects.

Python was selected as the programming language for development, offering platform independence and easy integration. PostgreSQL databases were chosen for customer data storage, while SQLite may serve to record system messages. The user interface design was excluded from this study's scope. Rather than custom web services, the ML-Flow Models API was utilized, leveraging its REST API. Linux distributions were favored for development because of API compatibility and stability, providing developers with additional tools and simplifying system administration.

6.3 Machine Learning Lifecycle

The machine learning life cycle involves the ideation, development, integration, and maintenance phases. Sections 1 and 2 cover idea generation, Sections 4 and 5 discuss development, and Section 6 addresses the details of integration and maintenance.

After a certain period, prediction accuracy decreased significantly, requiring retraining, testing, and commissioning of machine learning models with degraded performance. Triggers in database management systems can identify models with reduced performance, allowing timely interventions. To reduce associated development costs, the open-source ml-flow library was utilized, which defines the foundation of the machine learning life cycle. It tracks experiments, facilitates collaborative code usage, and provides unified control over various ML libraries. The project architectures vary, but the implemented architecture separates the experimental work and the model packaging, as depicted in Fig. 1, with ML flow connecting model performance logs and metrics between these phases.

Although the initial costs associated with developing and maintaining the entire architecture are higher than those of a standalone

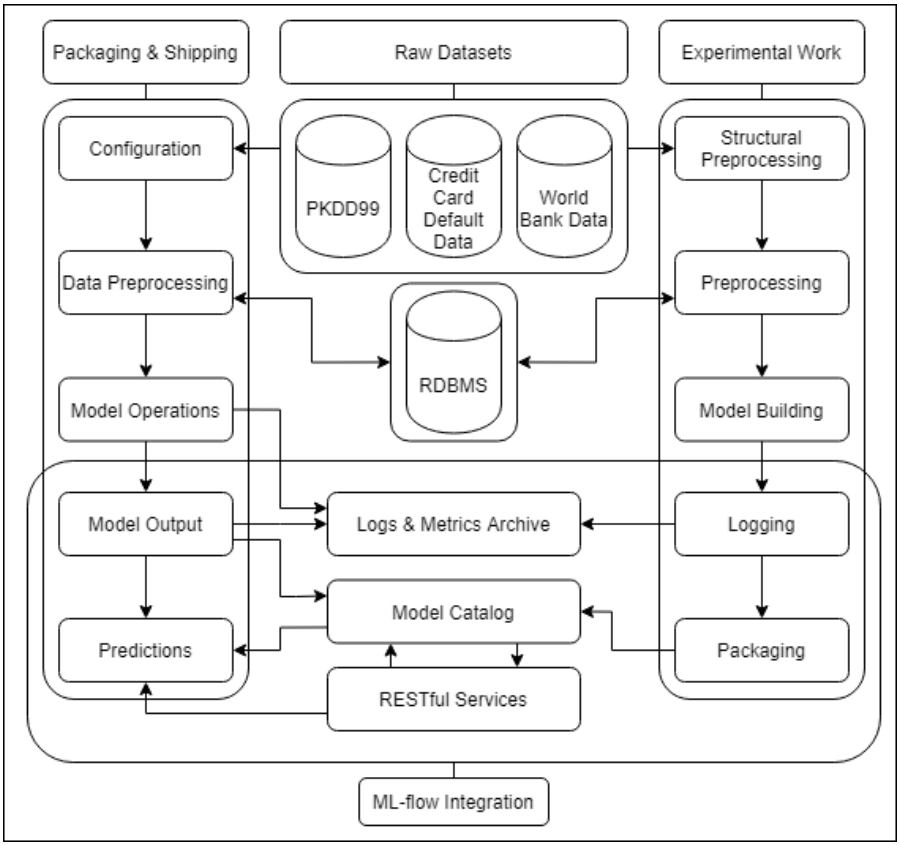

Figure 1. Project architecture

system, we remain optimistic about the long-term advantages gained from increased precision in detecting credit card churn and applying preventative measures. Despite upfront costs, the improved efficiency of this method is expected to generate significant profits for financial institutions in the future, effectively balancing out operational costs.

7 Conclusions

This work was carried out to show the different aspects of data processing, model training, and management methods that researchers use to study customer behavior and value in the banking and financial services industry. The results of this work showed that the problem should be clearly defined and the differences between other industries (telecommunications, public services, etc.) should be determined in order to complete the integration and structuring of the data in a proper way. It is crucial to thoroughly investigate the causes behind the loss of data in order to address the gaps. It is important to distinguish between losses resulting from invalid values due to logical operations and losses arising from customers' concerns about sharing information. It is essential to examine samples classified incorrectly, as sometimes misclassification occurs due to insufficient critical data, as demonstrated in this study, rather than solely relying on model parameters. To mitigate these losses, it is essential to devise suitable data recovery strategies based on expert opinions.

Another approach is to use synthetic data sets that are adapted to the real world. Although models trained with synthetic data may offer short-term benefits, their static nature can pose long-term problems. Therefore, utilizing model feedback and modern AI algorithms can provide dynamic structure to static synthetic datasets used for training churn models.

To reduce the prediction period to less than three months, which is considered short-term in this study, future investigations may focus on employing advanced deep learning algorithms such as long short-term memory (LSTM) networks. These algorithms can be applied to operational data from PKDD 99 and similar datasets. The integration of modern technologies such as contemporary Time Series Database (TSDB) systems and message broker technologies could greatly facilitate the implementation of predictive models that adapt to the environment in near real-time. Exploring how these findings can contribute to the development of more sophisticated churn prediction models and methodologies, particularly by leveraging LSTMs and near-real-time message brokers, presents an exciting avenue for further research.

Acknowledgements

We would like to thank the intermediary research organization that provided the data sets required for this study.

References

[1] A. M. Turing, *Computing Machinery and Intelligence*. Dordrecht: Springer Netherlands, 2009, pp. 23–65. [Online]. Available: https://doi.org/10.1007/978-1-4020-6710-5_3

[2] R. J. Solomonoff, "An inductive inference machine," in *IRE Convention Record, Section on Information Theory*, vol. 2, 1957, pp. 56–62.

[3] M. Zaharia, A. Chen, A. Davidson, A. Ghodsi, S. A. Hong, A. Konwinski, S. Murching, T. Nykodym, P. Ogilvie, M. Parkhe *et al.*, "Accelerating the machine learning lifecycle with mlflow." *IEEE Data Eng. Bull.*, vol. 41, no. 4, pp. 39–45, 2018.

[4] F. Altınışık and H. H. Yılmaz, "Predicting customers intending to cancel credit card subscriptions using machine learning algorithms: A case study," in *2019 11th International Conference on Electrical and Electronics Engineering (ELECO)*. IEEE, 2019, pp. 916–920.

[5] F. F. Reichheld and W. E. Sasser, "Zero defeofions: Quoliiy comes to services," *Harvard business review*, vol. 68, no. 5, pp. 105–111, 1990.

[6] G. Nie, W. Rowe, L. Zhang, Y. Tian, and Y. Shi, "Credit card churn forecasting by logistic regression and decision tree," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15 273–15 285, 2011.

[7] I.-C. Yeh and C.-h. Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2473–2480, 2009.

[8] D. Van den Poel and B. Lariviere, "Customer attrition analysis for financial services using proportional hazard models," *European journal of operational research*, vol. 157, no. 1, pp. 196–217, 2004.

[9] C. Zopounidis, M. Mavri, and G. Ioannou, "Customer switching behaviour in greek banking services using survival analysis," *Managerial Finance*, 2008.

[10] A. Lemmens and C. Croux, "Bagging and boosting classification trees to predict churn," *Journal of Marketing Research*, vol. 43, no. 2, pp. 276–286, 2006.

[11] C.-S. Lin, G.-H. Tzeng, and Y.-C. Chin, "Combined rough set theory and flow network graph to predict customer churn in credit card accounts," *Expert Systems with Applications*, vol. 38, no. 1, pp. 8–15, 2011.

[12] Y. Chen, L. Zhang, Y. Shi, and X. Liu, "Discovering intelligent knowledge for credit card churn management through second-order mining using multiple criteria linear programming," *International Information Institute (Tokyo). Information*, vol. 16, no. 11, p. 7941, 2013.

[13] Y. Chen, L. Zhang, and Y. Shi, "Post mining of multiple criteria linear programming classification model for actionable knowledge in credit card churning management," in *2011 IEEE 11th International Conference on Data Mining Workshops*. IEEE, 2011, pp. 204–211.

[14] B. He, Y. Shi, Q. Wan, and X. Zhao, "Prediction of customer attrition of commercial banks based on svm model," *Procedia Computer Science*, vol. 31, pp. 423–430, 2014.

[15] J. Xiao, Y. Xiao, A. Huang, D. Liu, and S. Wang, "Feature-selection-based dynamic transfer ensemble model for customer churn prediction," *Knowledge and information systems*, vol. 43, no. 1, pp. 29–51, 2015.

[16] R. Rajamohamed and J. Manokaran, "Improved credit card churn prediction based on rough clustering and supervised learning techniques," *Cluster Computing*, vol. 21, no. 1, pp. 65–77, 2018.

[17] E. R. Altman, "Synthesizing credit card transactions," *arXiv preprint arXiv:1910.03033*, 2019.

[18] D. Pehlivanli, S. Eken, and E. Ayan, "Detection of fraud risks in retailing sector using mlp and svm techniques," *Turkish Journal*

of Electrical Engineering & Computer Sciences, vol. 27, no. 5, pp. 3633–3647, 2019.

[19] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[20] A. G. Bluman, *Elementary statistics: A step by step approach: A brief version*. McGraw-Hill, 2013, no. 519.5 B585E.

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[22] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[23] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[24] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, 1974.

[25] C. D. Manning, P. Raghavan, and H. Schutze, "Introduction to information retrieval," *Ch*, vol. 20, pp. 405–416, 2008.

Fehim Altınışık, Ahmet Sayar

Received April 3, 2023
Revised April 8, 2024
Revised July 20, 2024
Accepted July 21, 2024

Fehim Altınışık
ORCID: https://orcid.org/0000-0003-1333-6411
Kocaeli University, Department of Computer Engineering.
Kabaoğlu, Baki Komşuoğlu Avenue 41001 İzmit Kocaeli/Turkey.
E–mail: fehim.altinisik@gmail.com

Ahmet Sayar
ORCID: https://orcid.org/0000-0002-6335-459X
Kocaeli University, Department of Computer Engineering.
Kabaoğlu, Baki Komşuoğlu Avenue 41001 İzmit Kocaeli/Turkey.
E–mail: ahmet.sayar@kocaeli.edu.tr