**Customer Churn Prediction Research Project Applying MLOps**

Ngoc Quang Vinh Pham

25100660

[*Ngoc.Q.Pham-1@student.uts.edu.au*](mailto:Ngoc.Q.Pham-1@student.uts.edu.au)

**I. Abstraction**

In the banking sector, retaining customers is significantly more cost-effective than acquiring new ones, making customer churn prediction crucial for proactive retention strategies. This research explores the complexities of customer behavior in the Vietnamese banking industry by using real-world transactional data and behavioral insights to gain an in-depth understanding of customer segments and churn drivers. While machine learning models have shown substantial promise in predicting churn, the operational challenge lies in deploying and managing these models at scale in real-time environments. This project employs advanced Machine Learning Operations (MLOps) frameworks—Kubernetes, Kubeflow, and Feast—to automate and scale the churn prediction pipeline, using Kubernetes for container orchestration, Kubeflow for lifecycle management, and Feast for feature consistency. By leveraging real-world data from a Vietnamese bank, this study seeks to deliver a robust, scalable, and dynamic solution that provides insights into the nuanced patterns of customer churn, enabling banks to manage retention efforts more effectively.

**II. Introduction**

1. **Background and Context**

In the competitive landscape of Vietnamese banking, customer churn prediction is vital for maintaining profitability and nurturing long-term relationships. Churn prediction involves identifying customers at risk of discontinuing the use of banking services, which might entail closing accounts, shifting to competitors, or reducing product engagement. With the capability to accurately anticipate churn, banks can implement targeted retention measures, offering tailored incentives to retain high-value customers. However, the real challenge lies in interpreting complex customer behaviors within diverse segments and integrating these insights into predictive models.

Customer behavior in the banking sector is intricate, driven by a range of factors such as financial habits, demographic characteristics, and service interactions. The Vietnamese banking environment presents an additional layer of complexity due to diverse customer segments, regional differences, and a rapidly evolving digital landscape. Given the wealth of behavioral data available—including transaction histories, account usage, and service interactions—this research emphasizes the need for a deep, data-driven approach to customer segmentation and behavioral analysis to improve churn prediction accuracy.

However, while machine learning has proven effective in predicting churn, deploying these models presents significant operational challenges. Banks must handle sensitive customer data in compliance with strict data privacy laws, while also addressing the dynamic nature of customer behavior. Traditional machine learning model deployments tend to rely on manual processes, which are not scalable and make retraining and updating models to reflect real-time trends difficult.

The adoption of MLOps (Machine Learning Operations) practices, which bring DevOps principles to machine learning, provides a solution to these challenges by automating much of the machine learning lifecycle—model development, deployment, monitoring, and retraining[1]. By using MLOps, organizations can improve scalability, reliability, and collaboration across teams, ensuring that models remain efficient and easy to maintain in production environments.

1. **Motivation**

As customer expectations and behaviors become more multifaceted, Vietnamese banks must adapt by developing sophisticated methods to capture these behaviors accurately. Traditional churn prediction models struggle to reflect the complexity of modern customer behavior, especially without an automated pipeline that can dynamically retrain and adapt based on evolving patterns. The integration of MLOps frameworks allows for a scalable, automated approach to deploying machine learning models that accurately capture and respond to customer behavioral nuances.

This research investigates how MLOps tools—specifically Kubernetes, Kubeflow, and Feast—can streamline the deployment and management of customer churn prediction models, while enabling a nuanced analysis of customer segments and behavioral patterns. By exploring real-world data from a Vietnamese bank, this project aims to uncover deeper insights into the unique behaviors and characteristics of different customer segments, supporting the bank in managing retention strategies more effectively.

1. **Objective**

The primary objective of this research is to create an automated, scalable churn prediction pipeline capable of capturing complex behavioral patterns and segmentation insights. Using MLOps frameworks, the specific goals are to:

Automate the training, deployment, and retraining of machine learning models that analyze customer segmentation and behavior for churn prediction.

Enhance the scalability and robustness of model deployment with Kubernetes and Kubeflow.

Maintain consistency in data handling and feature engineering with Feast, ensuring accuracy across both training and real-time inference.

Enable Vietnamese banks to develop informed retention strategies based on deep behavioral insights derived from real-world customer data.

1. **Scope**

This research is focused on applying MLOps frameworks to the specific context of customer churn prediction within a Vietnamese bank. The methods used—segmentation analysis, behavioral profiling, and MLOps deployment—can be generalized to other financial institutions and industries where customer behavior insights are valuable. Rather than developing new machine learning algorithms, this project seeks to refine the operationalization of existing models to better reflect the complexity of customer behavior, enhancing model scalability and real-time responsiveness.

**III. Literature Review**

First and foremost, we explored academic publications, industry reports, and existing technologies related to customer churn prediction and MLOps frameworks. There is an abundance of research in both fields, and many studies have been highly informative and practical, offering inspiration for this research. A common theme among the studies is the increasing complexity of customer behavior and the rapid evolution of machine learning technologies, which have highlighted the need for scalable, automated solutions to efficiently handle customer churn prediction at scale, especially in the banking industry.

One such study, "Customer Churn Prediction in the Banking Sector Using Machine Learning-Based Classification Models" by Tran et al. (2023)[2], investigates how machine learning models such as Random Forest, Logistic Regression, and Support Vector Machine (SVM) can be used for churn prediction in the banking sector. The authors highlight that Random Forest performed the best with a 97% accuracy, showcasing its ability to handle complex banking data. This study also explored the role of customer segmentation, concluding that it did not significantly affect model accuracy, but the choice of machine learning models played a crucial role in prediction performance.

Similarly, the article "Deployment of ML Models using Kubeflow on Different Cloud Providers" (2022)[3] focuses on the operational challenges that arise in deploying machine learning models in production environments. The authors argue that as businesses generate increasingly large datasets, manual model management quickly becomes inefficient, particularly when regular model retraining and performance monitoring are required. Their research explores how Kubernetes and Kubeflow have been applied to automate the deployment, scaling, and retraining of machine learning models across different cloud platforms. By leveraging containerization and orchestration, the authors demonstrate that MLOps frameworks like Kubeflow and Kubernetes can substantially reduce the time and effort needed for operationalizing machine learning models while also ensuring consistency in model performance.

Furthermore, "How Feature Stores Enhance Model Performance in ML Technology" (2023)[3] explores the pivotal role of feature stores in machine learning pipelines. The authors emphasize that a feature store serves as a centralized repository that simplifies feature engineering, ensuring that the same features used during model training are consistently applied during inference. This consistency helps improve model performance, reduce errors, and streamline the machine learning workflow by enabling faster experimentation and feature reuse across projects. By facilitating collaboration between data engineers and data scientists, feature stores also help ensure that high-quality, well-engineered features are accessible for real-time and batch processing, ultimately improving the reliability of machine learning models.

All these studies support the integration of MLOps frameworks to streamline the deployment and management of machine learning models for churn prediction. Tran et al. emphasize the importance of machine learning algorithms such as Random Forest in banking churn prediction, while Kreuzberger et al. highlight the operational efficiencies gained from Kubernetes and Kubeflow. The article on feature stores demonstrates the importance of feature management for maintaining consistency across the machine learning pipeline, particularly when dealing with large datasets and real-time predictions.

The common thread across these studies is the recognition that as customer behavior becomes increasingly complex and data-driven, traditional machine learning workflows are insufficient for handling the scale and velocity of data. By integrating MLOps practices, organizations can automate and scale the entire machine learning lifecycle, from feature engineering to model deployment and monitoring. In this research, we aim to expand on these findings by creating a robust, automated pipeline that leverages the full potential of Kubernetes, Kubeflow, and Feast, providing a more efficient and scalable solution for bank customer churn prediction.

**IV. Problem Statement**

In the modern Vietnamese banking sector, customer retention is crucial to sustaining profitability and staying competitive. The challenge lies in accurately predicting which customers are likely to leave, based on their unique behaviors, segments, and interactions. Traditional churn prediction methods often fall short of capturing the full complexity of customer behavior, particularly without continuous retraining and real-time monitoring. In a real-world context, such as that of a Vietnamese bank, customer behaviors are influenced by diverse factors including economic conditions, personal finance habits, and cultural aspects.

While machine learning models such as **Random Forests**, **Gradient Boosting Machines**, and **Support Vector Machines** are highly effective at predicting churn based on customer transaction data, deploying these models in real-time production environments is complex. Banks handle sensitive data and must comply with strict privacy regulations, adding another layer of complexity to managing these models. Additionally, customer behaviors shift rapidly, meaning churn prediction models need to be frequently retrained and updated.

Manual retraining is often slow and inefficient, which can lead to models becoming outdated and less accurate. Moreover, ensuring consistent feature management across training and production environments is critical to maintaining model accuracy and compliance with data governance standards. Without an automated system, the risk of errors increases, and the models may fail to keep pace with real-time customer behavior.

This research seeks to address the critical challenges of **automating the deployment, scaling, and retraining** of churn prediction models in the banking sector and the **challenges in capturing nuanced behavioral patterns and customer segmentation**. By leveraging **MLOps frameworks**—specifically **Kubernetes**, **Kubeflow**, and **Feast**—the goal is to build a scalable and automated solution that can handle large datasets and adapt to shifting customer behaviors. These tools will enable banks to deploy and manage machine learning models efficiently, ensuring **operational efficiency**, **data consistency**, and **regulatory compliance**.

The main challenges this research aims to solve include:

* **Complexity in customer behavior:** How can MLOps frameworks support models that adapt to multifaceted customer behaviors and segmentation insights?
* **Scalability and automation:** How can Vietnamese banks handle large, dynamic datasets and ensure real-time responsiveness without compromising security or regulatory compliance?
* **Feature consistency:** How can feature stores like Feast ensure consistent and accurate data handling across training and production environments?
* **Data privacy compliance:** How can banks operationalize MLOps frameworks while adhering to strict data privacy laws and ensuring data governance?

By addressing these challenges, the research aims to provide banks with a comprehensive solution that improves customer retention strategies while enhancing the efficiency of machine learning operations.

**V. Methodology**

The methodology for this research is designed to build an end-to-end MLOps pipeline for predicting customer churn in the banking sector. The key components include data acquisition, model selection, MLOps framework setup, and evaluation. The pipeline will automate the entire machine learning lifecycle, from data ingestion and model training to deployment, monitoring, and retraining. Below is a detailed breakdown of the methodology.

1. **Data Collection and Preprocessing**

**Data Collection:**

The data for this project will consist of transactional and behavioral records from a bank’s customer database. Key data points will include:

* Customer demographics (age, income, location)
* Transaction history (deposits, withdrawals, credit card usage)
* Account behavior (account activity, loan defaults, number of financial products)
* Customer service interactions (complaints, inquiry response times)

**Plan for Completion:**

We are in the process of finalizing the dataset and applying additional feature engineering techniques. This section will be expanded with more detailed preprocessing steps.

**Data Preprocessing:**

To ensure data quality and compatibility with the machine learning models, the following preprocessing steps will be applied:

* Handling Missing Data: Missing values in key fields such as income or transaction amounts will be imputed using statistical methods (mean/mode imputation).
* Feature Engineering: New features will be engineered to capture important behavioral signals, such as account inactivity periods, transaction volume trends, and loan repayment behaviors. These features will later be managed through Feast.
* Normalization and Scaling: Financial data such as balances and transaction amounts will be normalized to prevent large values from disproportionately influencing the model.
* Data Splitting: The dataset will be split into training (70%) and testing (30%) subsets, ensuring a balanced distribution of churn and non-churn cases across both sets.

1. **Exploratory Data Analysis**

(Placeholder)

This section will include a detailed analysis of the dataset using visualizations (e.g., **histograms**, **correlation matrices**) and key descriptive statistics.

**Planned next steps**: This analysis will be added once the dataset is fully prepared and initial preprocessing is completed.

1. **Model Selection**

Several machine learning models will be developed and tested to predict bank customer churn. Each model will be evaluated based on its ability to identify churners from the bank’s customer base using a combination of transactional and behavioral data.

1. Logistic Regression:

This interpretable linear model will serve as a baseline for performance comparison. Logistic regression will be used to model the binary outcome of churn vs. non-churn, providing an initial benchmark against which the more advanced models will be assessed.

1. Random Forests:

Random Forest, an ensemble learning method, will be used for its ability to handle both numerical and categorical data efficiently. It is particularly effective for churn prediction, as it can capture complex interactions between variables such as transaction history and customer engagement. Random Forest’s built-in feature importance metrics will also be leveraged to identify the key drivers of churn in the banking sector.

1. Extreme Gradient Boost:

Given its performance in handling structured datasets and imbalanced data, XGBoost will be used to further enhance churn prediction accuracy. This gradient-boosting algorithm optimizes decision trees to minimize errors in predicting churn. Its ability to handle skewed data distributions (e.g., far fewer churners than non-churners) makes it ideal for the bank churn prediction context, where churn events may be relatively rare.

1. Neural Networks (Optional):

In cases where deep insights into customer behavior are required, Neural Networks may be employed. These models are useful for capturing non-linear relationships between variables but will be balanced against interpretability and computational cost, which are important in banking due to the need for regulatory compliance and transparency.

Hyperparameter tuning for each model will be automated using Katib, a tool within Kubeflow. Katib will perform random search and Bayesian optimization to fine-tune parameters such as learning rate and tree depth, ensuring optimal model performance.

1. **Model Evaluation**

(Placeholder)

Once the models are trained, this section will include an evaluation of model performance using various metrics. A confusion matrix, precision-recall curve, and other visualizations will be added to interpret model effectiveness.

Planned next steps: Evaluation will be added once the model training is completed.

1. **MLOps Framework Implementation**

The MLOps pipeline for the bank churn prediction model will be built using several key components, with Google Cloud being the central platform due to the limited capacity of the local machine. Google Cloud provides the necessary scalability, compute power, and infrastructure needed to manage resource-intensive machine learning operations that cannot be efficiently handled locally.

**1. Kubernetes for Orchestration:**

Kubernetes will be used for orchestrating the containerized machine learning models. These models will be packaged into Docker containers to ensure consistency between development and production environments. However, the latest versions of Kubernetes have transitioned the default container runtime from Docker Engine to Containerd, which offers improved performance and resource management. Given the high computational needs of real-time churn prediction, deploying Kubernetes on Google Kubernetes Engine (GKE)—Google Cloud's managed Kubernetes service—will provide the necessary scalability and robustness.

Since the local machine lacks the necessary capacity to efficiently handle a production-scale Kubernetes setup, Google Cloud provides a much more feasible solution by offering elastic resource allocation, automatic scaling, and high availability for the machine learning models.

**2. Kubeflow for Pipeline Automation on Google Cloud:**

Due to the limited local computational capacity, Kubeflow will be deployed on Google Cloud for efficient pipeline automation. Kubeflow is an open-source MLOps platform that integrates seamlessly with Kubernetes and automates the various stages of the machine learning lifecycle[6]. This includes:

* Model Training:

The training of machine learning models on large datasets is computationally intensive, especially when fine-tuning and retraining are required. Google Cloud’s GKE will provide the necessary compute resources, enabling Kubeflow to automate the training process and retrain models as new customer data becomes available. This prevents performance bottlenecks that would otherwise occur if the pipeline were deployed on a local machine.

* Model Validation:

Kubeflow will automatically validate models against predefined performance benchmarks. Google Cloud's scalable infrastructure ensures that even large models or those requiring significant compute (e.g., XGBoost and Random Forest) are validated efficiently.

* Model Deployment:

Kubeflow will deploy trained models into production environments on GKE. By leveraging Google Cloud’s managed Kubernetes service, the deployment can scale automatically to handle increased traffic or data loads, something a local machine would struggle to accommodate.

* Monitoring and Retraining:

Kubeflow’s monitoring tools will track model performance in real time. If model accuracy or other key performance metrics degrade over time, Kubeflow will automatically trigger retraining using the resources available on Google Cloud, ensuring that the models remain accurate without manual intervention.

**3. Feast as a Feature Store on Google Cloud:**

Feast will be used as a centralized feature store to manage the features used during model training and inference. This includes managing features such as customer transaction history, credit score trends, and account activity. Due to the large volumes of data and the need for real-time feature updates, the limited local infrastructure is insufficient for managing these tasks efficiently. By deploying Feast on Google Cloud, the following advantages are achieved:

* Feature Consistency:

Feast ensures that the same features are used in both batch training and real-time inference, preventing training-serving skew. By using Google Cloud, Feast can scale to handle the growing feature set without performance degradation, a challenge that would arise if the feature store were run on a local machine.

* Batch and Real-Time Data Management:

Google Cloud allows Feast to handle both batch data (historical transaction data for model training) and real-time data (new transactions or customer updates). This ensures that the churn prediction models always operate with the most up-to-date customer information, enabling banks to make accurate predictions in real time.

* Feature Versioning and Governance:

Google Cloud offers the necessary resources for managing feature versions and maintaining governance over the entire feature pipeline. This is critical in the banking industry, where compliance with data regulations such as GDPR is mandatory. By utilizing Feast on Google Cloud, all feature updates, transformations, and usage are tracked and versioned, ensuring full compliance.

**Current status:** The pipeline setup on Google Cloud has been initiated. The implementation is in progress, with Kubernetes and Kubeflow being integrated to manage model deployment and scaling. Feast is being integrated for real-time feature management, ensuring feature consistency between training and production.

**Planned next steps:** Finalize the integration of Feast and complete the pipeline automation using Kubeflow. This section will be expanded with more technical details, including how Docker containers are used and how GKE enables scalability.

1. **Compliance and Security**

(Placeholder)

**VI. Results & Discussion**

(Placeholder)

The results will be added after the model evaluation, along with a discussion of the findings and the impact of MLOps frameworks on model performance and scalability.

**VII. Conclusion**

(Placeholder)

The final conclusions will summarize the outcomes of the research and highlight the benefits and limitations of using MLOps frameworks in the banking sector.

**VIII. Future Work**

(Placeholder)

Future work will focus on exploring further optimizations of the pipeline, extending the methodology to other sectors, and improving real-time model adaptation through more advanced MLOps tools.

**VIII. References**

[1] Goyal, "A Comprehensive Guide on MLOps for Machine Learning Engineering," Analytics Vidhya, Mar. 2022. Available: <https://www.analyticsvidhya.com/blog/2022/03/a-comprehensive-guide-on-mlops-for-machine-learning-engineering/>

[2] H. D. Tran, N. Le, and V. Nguyen, "Customer Churn Prediction in the Banking Sector Using Machine Learning-Based Classification Models," Interdisciplinary Journal of Information, Knowledge, and Management, vol. 18, pp. 87-105, 2023.

Available:

<https://www.informingscience.org/Publications/5086>

[3] D. Kreuzberger, N. Kühl, and S. Hirschl, "Deployment of ML Models using Kubeflow on Different Cloud Providers," arXiv preprint, May 2022.

Available:

<https://ar5iv.labs.arxiv.org/html/2206.13655>

[4] P. Klushin, "How Feature Stores Enhance Model Performance in ML Technology," Future with Tech, Nov. 2023.

[5] Kubernetes “Changing the Container Runtime on a Node from Docker Engine to containerd” July. 2023.

Available:

<https://kubernetes.io/docs/tasks/administer-cluster/migrating-from-dockershim/change-runtime-containerd/>

[6] Kubeflow Definition - Wikipedia

Available:

<https://en.wikipedia.org/wiki/Kubeflow>