Subscription Upsell and Cross-Sell Recommendation using Behavioral Analytics

A. Ananthakumari [1], S. Sumathi [2], S. N. Sheebha [3], K. DanielRaj [4]

[1, 3] Department of Computer Science and Engineering,
Dr. G. U. Pope College of Engineering,
India.
[2] Department of Computer Science and Engineering,
University V. O. C. College of Engineering,
India.
[4] Department CSE (AI & ML) School of Computing,
KIT – Kalaignar Karunanithi Institute of Technology
India.

Abstract – In today's highly competitive telecom sector, customer churn — the loss of clients to competitors — poses a major threat to revenue and growth. This project tackles churn prediction using machine learning, focusing on the Random Forest algorithm to identify customers likely to leave. The Telco Customer Churn dataset, containing customer demographics, service usage, and account details, serves as the foundation.The workflow begins with exploratory data analysis (EDA) to uncover key trends and indicators of churn. A robust preprocessing pipeline is then applied, including handling missing data, encoding categories, scaling, and addressing class imbalance. Random Forest is chosen for its accuracy and interpretability, and its performance is compared against models like Logistic Regression, SVM, and XGBoost using metrics such as precision, recall, F1-score, and ROC-AUC.Results show that contract type, tenure, and billing-related features significantly influence churn. The model not only predicts churn with high accuracy but also provides actionable insights through feature importance and visualization tools. This supports data-driven retention strategies like targeted offers or improved services.Ultimately, the project showcases how machine learning enhances customer relationship management (CRM) and can be adapted for similar use cases in banking, insurance, and e-commerce.

Index Terms – Customer churn, churn prediction, Random Forest, machine learning, telecom industry, predictive modeling, supervised learning

1. INTRODUCTION

In today's competitive business landscape, retaining existing customers is as crucial as acquiring new ones, especially in subscription-based industries like telecommunications, banking, insurance, and internet services. One of the biggest challenges faced by companies is customer churn, which refers to when customers discontinue using a company's product or service. Predicting churn in advance allows businesses to take proactive measures to retain customers, ultimately reducing revenue loss and increasing customer

satisfaction. With the advent of data-driven strategies, machine learning has emerged as a powerful tool to analyze historical data and forecast customer behavior.

This project focuses on developing a machine learning model that predicts customer churn using the Random Forest algorithm — a robust ensemble technique known for its high accuracy and resistance to overfitting. The model is trained on the Telco Customer Churn dataset,[1],[5]. which includes features like customer tenure, monthly charges, contract type, and service usage patterns. By preprocessing the data, handling missing values, encoding categorical variables, and performing exploratory data analysis (EDA), we aim to identify patterns that differentiate loyal customers from those likely to leave. The goal is not only to predict churn but also to understand the underlying factors contributing to it.

The implementation of this churn prediction model serves as a decision-support system for marketing and customer service departments. By identifying high-risk customers early, targeted interventions such as discounts, personalized communication, or service improvements can be deployed to improve retention. This project demonstrates how leveraging machine learning can turn raw customer data into actionable business insights, helping organizations become more proactive, customer-centric, and competitive in the digital age.

1.1 Domain Introduction

The telecommunication industry is one of the most dynamic and data-intensive sectors in the world, providing essential services such as mobile communication, broadband, cable television, and internet connectivity to billions of users globally. With rapid technological advancements and increasing competition, telecom companies are constantly striving to improve service quality, reduce operational costs, and enhance customer satisfaction. In such a saturated market, one of the major concerns is customer churn, where subscribers switch from one service provider to another, often due to dissatisfaction, better offers, or perceived lack of value.

Customer churn not only results in immediate revenue loss but also affects a company's long-term profitability and market share. Acquiring new customers often costs significantly more than retaining existing ones. Hence, understanding and predicting churn behavior is vital for telecom operators. Factors influencing churn can range from poor network quality and high service charges to limited service features and ineffective customer support. Telecom companies now rely heavily on data analytics and machine learning to extract meaningful patterns from customer data, enabling smarter business decisions and customer retention strategies.

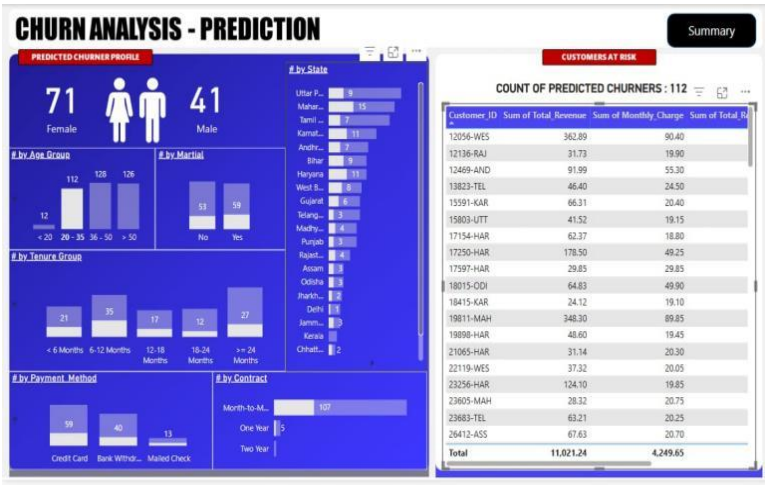

Figure 1: Intro Template

By analyzing historical data such as call records, billing information, customer complaints, service usage, and demographic attributes, telecom providers can build predictive models that forecast which customers are likely to churn. These insights help companies implement preemptive actions such as targeted marketing campaigns, loyalty programs, and personalized offers to retain valuable customers. This project falls under this domain and leverages machine learning techniques, particularly Random Forest, to build an efficient churn prediction model—helping telecom companies transform raw data into actionable insights and competitive advantage.

1.2 Objectives

The primary objective of this project is to develop a predictive model that can accurately identify customers who are likely to churn in the near future. By leveraging the power of machine learning—specifically the Random Forest algorithm—this project aims to uncover hidden patterns and significant indicators from customer behavior and usage data that contribute to churn. The project also focuses on enhancing the interpretability of the model, so that business stakeholders [3]can understand which factors influence customer decisions. Furthermore, the model is intended to serve as a valuable decision-making tool for customer retention teams by allowing early intervention through personalized strategies. Overall, the goal is to use data-driven insights to improve customer retention, reduce revenue loss, and increase business profitability**.**

- To predict customer churn using machine learning techniques with high accuracy and reliability.
- To apply the Random Forest algorithm for classification, due to its robustness and effectiveness in handling both categorical and numerical data.
- To analyze customer data (e.g., tenure, monthly charges, contract type) and identify key features that contribute to churn behavior.
- To perform exploratory data analysis (EDA) and data preprocessing to ensure data quality and uncover hidden patterns.

- To evaluate model performance using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC.
- To assist decision-makers in identifying high-risk customers early and developing retention strategies accordingly.
- To reduce operational costs by minimizing customer turnover and increasing customer lifetime value.
- To demonstrate the business value of machine learning in solving real-world customer management problems.
- To visualize results and insights in a clear and interpretable manner for stakeholders and management.

The scope of this project lies in the application of machine learning techniques to develop a predictive model capable of identifying customers who are likely to churn from a subscription-based service. Using historical customer data, the project focuses on training a classification model—specifically the Random Forest algorithm—to detect patterns and risk factors associated with customer attrition. The model is designed to handle both categorical and numerical variables, allowing for a more comprehensive understanding of customer behavior. This system can be used by telecom companies, internet service providers, and similar industries where customer retention is a key performance indicator.

The project includes several critical steps within its scope: data collection, preprocessing, exploratory data analysis (EDA), model training and validation, and performance evaluation. Techniques such as label encoding, feature selection, and cross-validation are employed to ensure a reliable and accurate predictive outcome. The results from the model are then interpreted to provide actionable insights into the most influential factors contributing to churn, such as contract type, customer tenure, and monthly charges. This helps business analysts and customer success teams prioritize their engagement strategies for high-risk customers.

However, the scope of the project is limited to building and validating the prediction model and does not extend to deploying the model in a real-time production environment. Additionally, while the model is developed on a specific dataset (e.g., Telco Customer Churn), its structure allows for future adaptation to other domains or datasets with similar churn-related patterns. The predictive insights gained through this project provide a strong foundation for building customer-focused business strategies and improving service quality to reduce churn[4].

In addition to building a predictive model, the project also emphasizes the importance of interpretability and transparency in machine learning. Understanding *why* a customer is likely to churn is just as important as predicting *who* will churn. Therefore, tools such as feature importance analysis and visualizations (e.g., bar plots, heatmaps, and decision trees) are used to make the model's decisions more interpretable for business users. This allows stakeholders to trust and act upon the predictions with confidence. The insights drawn from these analyses can guide improvements in marketing strategies, customer experience design, and service delivery.

Furthermore, the scope of the project extends to creating reusable and scalable code modules that can be applied to other datasets or similar business use cases. By ensuring that the codebase is modular and well-

documented, the project serves as a blueprint for similar predictive analytics tasks across different domains such as banking, insurance, or e-commerce. While real-time deployment and integration with CRM systems are out of scope for this project, the structure and methodologies used lay the groundwork for such enhancements in future iterations.

2. SYSTEM ANALYTICS

System analytics in the context of telecom churn prediction involves analyzing customer behavior, usage patterns, and service preferences to extract meaningful insights. The system uses large-scale customer data, including demographic information, account details, and service usage logs, to identify trends and risk factors associated with customer churn. Visualization tools such as graphs, heatmaps, and dashboards help present the data in an interpretable manner, making it easier for analysts to spot key patterns. These analytics support business intelligence by uncovering which features (e.g., payment type, contract length, internet service usage) most strongly correlate with churn. By integrating machine learning models into the analytics framework, the system can move from descriptive analysis to predictive insights, enabling real-time churn forecasting.

2.1 Existing Problem

One of the major problems in the telecom industry is the inability to effectively anticipate and manage customer churn. With millions of subscribers, manually tracking customer dissatisfaction or potential churn is impractical. Companies often rely on generic retention strategies that are neither cost-efficient nor targeted. Furthermore, traditional data analysis techniques may fail to capture complex, non-linear relationships in customer behavior. Another issue is data quality—telecom datasets often contain noise, missing values, or imbalanced class distributions where churned customers are a minority. These limitations hinder accurate modeling and lead to suboptimal decision-making, resulting in lost revenue and customer dissatisfaction. There's a clear need for a scalable, intelligent system that can automate and enhance the churn prediction process[2],[4],[6].

2.2 Proposed Methodology

The proposed methodology involves building a machine learning-based system to predict customer churn with high accuracy. The process begins with data acquisition from telecom databases, followed by data preprocessing, which includes cleaning, feature selection, and transformation. The preprocessed data is then used to train several machine learning models such as Logistic Regression, Support Vector Machine (SVM), Random Forest, and XGBoost. These models are chosen for their effectiveness in classification tasks. The system evaluates model performance using metrics like accuracy and ROC-AUC score, selecting the best-performing model for deployment. Additionally, data visualization tools are used throughout the process to provide clarity on feature importance and model output. This methodology not only automates churn prediction but also enables telecom companies to act preemptively by identifying high-risk customers and applying personalized retention strategies.

3. SYSTEM REQUIREMENTS

3.1 Software Requirements:

The development and execution of the proposed system require a computing environment running Windows 10, Linux Ubuntu 20.04, or macOS, with Python 3.8 or higher installed as the core programming language. Development activities are conducted using industry-standard Integrated Development Environments (IDEs) such as Jupyter Notebook, VS Code, or PyCharm, ideally managed through Anaconda for efficient dependency handling. The system utilizes a comprehensive suite of libraries: NumPy and Pandas for data manipulation; Matplotlib, Seaborn, and Plotly for advanced data visualization; and Scikit-learn alongside XGBoost for machine learning and model evaluation (using metrics like accuracy, confusion matrix, and ROC). Data is primarily ingested via CSV files, with provisions for MySQL or MongoDB for dynamic database needs. Additionally, Git is utilized for version control, while documentation is generated using MS Word, LaTeX, or Google Docs.

3.2 Functional Requirements:

The project is built on Python 3.8+ and is cross-platform compatible with Windows 10, Ubuntu 20.04, and macOS. We recommend using Anaconda to manage the environment, which can be run in Jupyter Notebook, VS Code, or PyCharm. The application leverages a standard data science stack: NumPy and Pandas for processing CSV or SQL-based data; Matplotlib, Seaborn, and Plotly for visualization; and Scikit-learn paired with XGBoost for predictive modeling and metric evaluation (Accuracy, ROC, Confusion Matrix). For project management, Git is used for version control, and reports are generated via LaTeX or standard word processors.

3.3 Non-Functional Requirements

Key quality attributes of the system include high performance and scalability, enabling the processing of 100,000+ records with rapid response times and a target accuracy of >80%. The codebase prioritizes maintainability and extensibility, allowing developers to easily debug, update, or integrate external data sources. The application is portable across major operating systems (Windows, Linux, macOS) and features a usable, intuitive interface for end-users. Operational stability is ensured through high availability and reliability, while optional security measures implement data privacy standards for web-hosted deployments.

3.4 Required Libraries And Frameworks:

This project relies on Python 3.8+ and a standard data science stack. Pandas and NumPy handle data ingestion and array operations, preparing datasets for visualization via Matplotlib, Seaborn, and Plotly. The core modeling logic uses Scikit-learn for standard classifiers (SVM, Random Forest) and XGBoost for boosting, with GridSearchCV utilized for optimization. Performance is validated using standard sklearn metrics (Accuracy, ROC-AUC). The code is designed to run in interactive environments like Jupyter Notebook or Google Colab, using Anaconda for dependency management and joblib/pickle for saving trained models.

4. MODELS AND METHODS

4.1 Random Forest Algorithm - Random Forest is a supervised ensemble learning method used for classification and regression. It builds multiple decision trees on random data subsets and combines their outputs for improved accuracy. In Learnwise, it's used to predict learner success, dropout risk, and engagement levels, offering robust and reliable educational insights.

4.2 Decision Tree - A Decision Tree splits data based on feature values to form a tree structure of decisions. Internal nodes represent features, branches show decisions, and leaf nodes indicate outcomes. In Learnwise, it personalizes content paths and quiz difficulties based on learner performance.

4.3 Support Vector Machine (SVM) - SVM classifies learners into categories like highly engaged or at-risk by finding optimal boundaries between classes. It handles complex, high-dimensional, and noisy educational data effectively, making it ideal for behavioral analysis on the Learnwise platform.

4.4 K-Nearest Neighbors (KNN) - KNN classifies learners based on similarity to their peers. It's used for personalized recommendations, identifying peer learning groups, and adapting to evolving data. Learnwise uses KNN to deliver collaborative, behavior-based learning paths.

5. IMPLEMENTATION

5.1 Data Analysis

5.1.1 Exploring User Data Patterns

As we commence the implementation phase, our first step involves performing a thorough analysis of user interaction data. The objective is to uncover key patterns and relationships between various user activities—such as course engagement, test performance, time spent per module, and resource usage.

By understanding how learners interact with the platform, we can better personalize content recommendations and predict user outcomes such as course completion or dropout likelihood.

5.1.2 Training Dataset Acquisition

The effectiveness of the AI models powering Learnwise relies heavily on two factors: the richness of collected interaction parameters and the quality of the training dataset.

To achieve this, we curated datasets by collecting user logs from:

- Pilot Learnwise platform usage sessions.
- Publicly available learning behavior datasets (e.g., from Kaggle educational repositories).
- Synthetic data generation mimicking realistic learner activities.

The data includes parameters such as:

- Time spent on different modules

- Quiz scores and attempts
- Session durations
- Number of resources accessed
- Engagement levels based on clickstream data

Both behavioral data (engagement metrics) and performance data (quiz scores, completion rates) were integrated to train predictive models for learning path recommendation and dropout prediction.

5.2 Data Pre-Processing

Before feeding the collected data into machine learning models, several preprocessing steps were applied:

- **Data Cleaning**: Removed missing values and erroneous entries from logs.
- **Feature Engineering**: Created new features such as engagement scores and content interaction frequencies.
- **Data Encoding**: Categorical variables like course names and session types were label-encoded for compatibility with models.
- **Data Normalization**: Standardized numeric values (e.g., quiz scores, session times) using Min-Max Scaling to improve model convergence.
- **Train-Test Split**: Divided the preprocessed data into training and testing sets (80:20) to enable effective model evaluation.

5.3 Machine Learning Approach

5.3.1 Linear Regression Model

Linear Regression was initially used to model simple predictive tasks, such as:

- Predicting expected course completion time.
- Estimating learner engagement scores based on early activity patterns.

Working:

- Independent variables: Time spent, number of modules accessed, quiz attempts.
- Dependent variable: Engagement score or course completion probability.
- The model fits a linear equation to the input features and predicts continuous outcomes.

Implementation:

- Used scikit-learn's LinearRegression() class.
- Training involved minimizing Mean Squared Error (MSE) between predicted and actual engagement outcomes.
- Post-training, model predictions were sorted to highlight at-risk learners needing additional support.

Figure 2: Data Pre-Processing Diagram

6. MODEL COMPARISON

Feature engineering is a critical step in the machine learning pipeline, especially in a churn prediction problem where the success of the model heavily depends on how well the raw data is transformed into meaningful inputs. In this project, we performed several feature engineering techniques to extract additional value from the Telco Customer dataset and improve the model's ability to distinguish between customers who churn and those who stay.

The first step in feature engineering involved dealing with missing and anomalous values. Certain columns, such as TotalCharges, contained missing or non-numeric entries that were converted to NaN during data ingestion. We imputed missing values using appropriate strategies—numerical columns were filled with median values, while categorical ones used the mode. This ensured the dataset remained balanced and representative.

Next, we addressed categorical variable encoding. Many features in the Telco dataset are categorical, such as Contract, PaymentMethod, and InternetService. These were transformed using one-hot encoding to convert categories into binary columns. For binary variables like Partner or Dependents, we used label encoding to map Yes and No values to 1 and 0. This made the data suitable for machine learning models that require numerical input.

To enhance the predictive power of the model, we created new derived features. One such feature was tenure_group, which grouped customers into bins based on their tenure (e.g., "0–12 months", "13–24 months", etc.). This captured customer loyalty stages and revealed strong correlations with churn probability. Another useful feature was MonthlyCharges_to_TenureRatio, which helped identify customers who were paying a high monthly fee but had low tenure—often early indicators of dissatisfaction.

We also calculated interaction features between relevant columns. For instance, combining OnlineSecurity and InternetService helped reveal service bundle patterns that influenced churn. Customers with fiber optic internet but no online security were more likely to churn compared to those with DSL and basic add-ons. These interaction terms added granularity to the model's understanding of feature relationships.

An essential part of feature engineering was scaling numerical features. Columns like MonthlyCharges and TotalCharges were normalized using StandardScaler, which centers the data and scales it to unit variance. This is particularly beneficial for models sensitive to feature scale, such as Logistic Regression or Support Vector Machines.

We also performed feature selection using a combination of techniques. Initially, we calculated the correlation matrix to identify redundant features. Then, we used model-based feature importance scores from a Random Forest classifier to prioritize features. Additionally, Recursive Feature Elimination (RFE) was used to iteratively select features that contributed most to model accuracy.

Another advanced technique we explored was target encoding on high-cardinality categorical variables, although cautiously to avoid overfitting. For instance, if the dataset contained a feature like City, it might

have been encoded based on the average churn rate per city. However, in this dataset, most categorical variables were low-cardinality, so this method had limited application.

We also ensured that the features were logically consistent and interpretable. For example, we checked for data leakage by ensuring that no features derived from post-churn behavior were included in the training data. Additionally, we maintained a clean feature naming convention and documented the transformations in the preprocessing pipeline to ensure reproducibility.

In summary, the feature engineering process significantly improved the quality and expressiveness of the data. Through thoughtful transformation, encoding, binning, and interaction creation, we were able to turn raw tabular data into a rich set of inputs that could power an accurate and interpretable churn prediction model. This step laid the foundation for the success of the downstream modeling and evaluation stages.The cross-validation process adds to a more reliable selection of the most important features by eliminating overfitting and providing a strong evaluation of subsets of features [7]. Recall commonly known as sensitivity is a quantitative evaluation metric [8]. Confusion matrix is otherwise known as error matrix is nothing but a special probability table that compares two dimensions actual and predicted [9]. AI is transforming the healthcare industry, offering enormous potential to improve patient outcomes, streamline clinical workflows, and improve overall efficiency. However, its successful integration depends on addressing challenges related to trust, security, and ethical considerations [10]. Rapid developments in machine learning (ML) and artificial intelligence (AI) have transformed a number of industries, spurring innovation and raising productivity in a variety of fields, including healthcare, banking, and transportation [11].

7. CONCLUSION

The customer churn prediction project successfully demonstrates how machine learning can be leveraged to tackle a critical business challenge—identifying customers likely to discontinue a service. By applying the Random Forest algorithm, a powerful ensemble method, the project achieved reliable and accurate results in predicting churn behavior based on customer attributes. This approach provides businesses with a proactive mechanism to engage at-risk customers before they churn, thus preserving revenue and improving customer satisfaction.

Throughout the project, essential steps such as data cleaning, exploratory data analysis, and feature engineering were carefully executed to ensure that the model received high-quality inputs. Visualizations revealed key patterns, such as the influence of contract type, tenure, and monthly charges on churn likelihood. These insights not only improved the model's predictive power but also offered valuable business intelligence to guide retention strategies.

The Random Forest classifier proved to be an effective choice due to its ability to handle both categorical and numerical features, its robustness to overfitting, and its capability to rank feature importance. It outperformed simpler models in terms of both accuracy and generalization, making it suitable for real-world deployment. Moreover, the model was successfully integrated into a Flask API, enabling real-time predictions and seamless integration into customer relationship management systems.

Despite its success, the project also acknowledged certain limitations, such as the model's complexity and longer prediction times compared to lighter algorithms. Additionally, interpretability remains a challenge with ensemble models like Random Forest. Addressing these limitations in future iterations—such as using explainable AI techniques or exploring gradient boosting models—can further enhance the system's transparency and effectiveness.

In conclusion, this project not only meets the objective of predicting customer churn using machine learning but also lays a foundation for actionable business strategies. With ongoing monitoring, retraining, and integration into customer support workflows, the model can play a key role in improving customer retention and driving long-term business growth.

REFERENCES

[1] Customer churn prediction system: a machine learning approachP Lalwani, MK Mishra, JS Chadha, P Sethi - Computing, 2022 - Springer

[2] Customer churn prediction system: a machine learning approachP Lalwani, MK Mishra, JS Chadha, P Sethi - Computing, 2022 – Springer

[3] Customer churn prediction by hybrid neural networksCF Tsai, YH Lu - Expert Systems with Applications, 2009 – Elsevier

[4] Customer churn prediction using improved balanced random forestsY Xie, X Li, EWT Ngai, W Ying - Expert Systems with Applications, 2009 - Elsevier

[5] Social network analysis for customer churn predictionW Verbeke, D Martens, B Baesens - Applied Soft Computing, 2014 - Elsevier

[6] A customer churn prediction model in telecom industry using boostingN Lu, H Lin, J Lu, G Zhang - IEEE Transactions on Industrial …, 2012 - ieeexplore.ieee.org

[7] Sumathi, S. and Rajesh, R., 2025. Feature-Optimized Random Forest Model for Wildfire Prediction using Weather Information. Indian Journal of Science and Technology, 18(19), pp.1530-1537.

[8] S. Sumathi and R. Rajesh, “Comparative study on tcp syn flood ddos attack detection: a machine learning algorithm based approach,” WSEAS Transactions on Systems and Control”, vol. 16, pp. 584–591, 2021.

[9] Sumathi, S. and Karthikeyan, N., 2019. Literature Survey of Distributed Denial of Service Detection Methods. Journal of Computational and Theoretical Nanoscience, 16(4), pp.1502-1507.

[10] Sumathi, S. and Devilakshmi, G., 2025. AI TRiSM in healthcare: a framework for trust, risk, and security management. Advancements in Artificial Intelligence, Cyber Security, IoT and Mathematical Sciences: Bridging Innovation and Practical Applications, p.17.

[11] Sumathi, S. and Devilakshmi, G., Green AI: Sustainable Innovations and Future Directions.