

Work Division - Telecom Customer Churn Prediction Project

Team Members

Abdul Kuddus (M24DE3002)(G23AI2044)
Ritik Sharma (M24DE3065)(G23AI2024)
Mohit Mathur (M24DE3050)(G23AI2034)
Jojo Joseph (M24DE3041)(G23AI2100)

Task Distribution

Abdul Kuddus (M24DE3002)(G23AI2044)

Responsibilities:

Data Preprocessing & ANN Model Implementation

Tasks:

- Data Preprocessing
 - Handle missing values in total_rech_data and Internet Type
 - Remove unnecessary columns (night_pck_user, fb_user, etc.)
 - Handle outliers in arpu_4g, arpu_5g, and vol_5g
 - Validate data quality and consistency
- Feature Engineering
 - Create total_recharge feature (sum of total_rech_amt and total_rech_data)
 - Derive quarterly features (Quarter, Quarter of Joining)
 - Implement one-hot encoding for categorical features
 - Apply StandardScaler for numerical features
- Artificial Neural Network (ANN) Model
 - Design ANN architecture (Dense layers with Dropout)
 - Implement training pipeline with Early Stopping
 - Train model with appropriate hyperparameters
 - Evaluate model performance (Accuracy, Precision, Recall, F1, ROC-AUC)
 - Save trained model for deployment

Report Sections:

Data Preprocessing Methodology
Feature Engineering Approach
ANN Model Architecture and Training
ANN Model Performance Analysis

Ritik Sharma (M24DE3065)(G23AI2024)

Responsibilities:

Exploratory Data Analysis & 1D CNN Model Implementation

Tasks:

- Exploratory Data Analysis (EDA)
 - Perform comprehensive data exploration (shape, info, columns)
 - Analyze distribution of target variable (churn percentage)
 - Visualize feature distributions and correlations
 - Identify patterns and insights in the data
 - Analyze class imbalance (~4.57% churn rate)
- Data Insights & Visualization
 - Create visualizations for key features
 - Analyze customer behavior patterns
 - Document data characteristics and anomalies
 - Generate statistical summaries
- 1D Convolutional Neural Network (CNN) Model
 - Design 1D CNN architecture (Conv1D, MaxPooling, BatchNorm)
 - Reshape data for CNN input requirements
 - Implement training pipeline with Early Stopping
 - Train model with appropriate hyperparameters
 - Evaluate model performance (Accuracy, Precision, Recall, F1, ROC-AUC)
 - Save trained model for deployment

Report Sections:

Exploratory Data Analysis and Insights
Data Visualization and Pattern Recognition
1D CNN Model Architecture and Training
CNN Model Performance Analysis

Mohit Mathur (M24DE3050)(G23AI2034)

Responsibilities:

Business Context & Transformer Model Implementation

Tasks:

- Business Overview & Problem Definition
 - Document business context and churn prediction importance
 - Define project objectives and success metrics
 - Literature review on churn prediction challenges
 - Explain business impact of churn prediction
- Data Loading & Setup
 - Set up data reading from multiple sources (CSV, S3, Database)
 - Implement data loading functions with error handling
 - Generate data dictionary documentation
 - Create train-test split (80/20 ratio)
- Transformer Model
 - Implement Multi-Head Attention architecture
 - Design Transformer model for tabular data
 - Implement training pipeline with Early Stopping
 - Train and evaluate Transformer model
 - Evaluate model performance (Accuracy, Precision, Recall, F1, ROC-AUC)
 - Save trained model for deployment

Report Sections:

Introduction and Business Context

Data Collection and Sources

Transformer Model Architecture and Training

Transformer Model Performance Analysis

Jojo Joseph (M24DE3041)(G23AI2100)

Responsibilities:

TabNet Model & Comprehensive Analysis

Tasks:

TabNet Model (Google AI, 2021)

- Implement TabNet architecture for interpretable predictions
- Configure TabNet hyperparameters (n_d, n_a, n_steps, gamma)
- Train TabNet with attention-based feature selection
- Evaluate model performance (Accuracy, Precision, Recall, F1, ROC-AUC)
- Generate feature importance visualizations
- Save trained model for deployment

Feature Importance Analysis

- Analyze TabNet feature importance scores
- Identify top features driving churn predictions
- Visualize feature importance rankings
- Provide interpretability insights

Comprehensive Model Comparison

- Create unified evaluation framework for all 4 models
- Generate comparative performance charts (ANN, CNN, Transformer, TabNet)
- Analyze strengths and weaknesses of each approach
- Compare model performance on imbalanced dataset

Final Documentation & Recommendations

- Compile complete project summary
- Document best practices and lessons learned
- Provide business recommendations based on model insights
- Prepare deployment recommendations
- Create executive summary for stakeholders

Report Sections:

TabNet Model Architecture and Training

Feature Importance and Interpretability Analysis

Comprehensive Model Comparison

Business Insights and Recommendations

Conclusion and Future Work

Timeline

Start Date: Project Initiation

Milestones:

Milestone 1: Data Exploration & Preprocessing Complete (Ritik & Abdul)

Milestone 2: All Models Trained & Evaluated (Abdul: ANN, Ritik: CNN, Mohit: Transformer, Jojo: TabNet)

Milestone 3: Model Comparison & Analysis Complete (Jojo)

Milestone 4: Final Report & Documentation Complete (All Members)

End Date: Project Submission

Report Structure

Each team member will contribute their sections to a unified report with the following structure:

1. Introduction (Mohit)

Business Overview

Problem Statement

Project Objectives

Data Sources and Collection

2. Data Analysis (Ritik & Abdul)

Exploratory Data Analysis (Ritik)

Data Visualization and Insights (Ritik)

Data Preprocessing and Cleaning (Abdul)

Feature Engineering (Abdul)

3. Model Development (All - Each Person One Model)

ANN Model (Abdul)

1D CNN Model (Ritik)

Transformer Model (Mohit)

TabNet Model (Jojo)

4. Results and Analysis (Jojo)

Model Performance Comparison

Feature Importance Analysis

Business Insights and Recommendations

5. Conclusion (Jojo)

Summary of Findings

Recommendations

Future Work

Notes

Equal Distribution: Each member has approximately 25% of the total workload

Collaboration: Team members should collaborate on data handoffs between phases

Code Quality: All code should be well-documented with comments

Reproducibility: Ensure all experiments are reproducible with random seeds

Communication: Regular sync-ups to ensure smooth integration of work

Version Control: Use Git for version control and collaboration

Documentation: Each member maintains their section of the final report