E-ISSN: 2707-6644
P-ISSN: 2707-6636
Impact Factor (RJIF): 5.43
www.computersciencejournals.com/ijcpdm
IJCPDM 2025; 6(2): 103-111
Received: 10-05-2025
Accepted: 15-06-2025

Emmah Victor Thomas
Department of Computer Science, Rivers State University, Port Harcourt, Rivers State, Nigeria

Ordu Princewill Okey
Department of Computer Science, Rivers State University, Port Harcourt, Rivers State, Nigeria

Bennett Emmanuel O
Department of Computer Science, Rivers State University, Port Harcourt, Rivers State, Nigeria

Corresponding Author:
Emmah Victor Thomas
Department of Computer Science, Rivers State University, Port Harcourt, Rivers State, Nigeria

Deep neural network model for customer attrition forecast in a telecommunication company

Emmah Victor Thomas, Ordu Princewill Okey and Bennett Emmanuel O

DOI: https://www.doi.org/10.33545/27076636.2025.v6.i2a.118

Abstract
The loss of customers is becoming a significant challenge for telecom companies due to the high cost of acquiring new customers and the critical need to retain existing ones. This dissertation explores the importance of predicting customer attrition in the telecommunications sector using a deep neural network (DNN) model. The study highlights the crucial role of customer retention in a highly competitive market. The system was developed using historical data, preprocessing techniques, and a customized DNN architecture. The methodology followed a DevOps approach, encompassing the collection, integration, and preprocessing of diverse datasets, followed by the construction and optimization of the DNN model with five layers using stochastic gradient descent. The findings demonstrate the model's impressive accuracy, achieving 98.1% after 100 epochs, along with improved precision. The results underscore the DNN model's effectiveness in predicting churn, emphasizing the value of iterative refinement through multiple training cycles. This research offers valuable insights and practical methodologies for telecom companies aiming to adopt proactive strategies to enhance customer retention and satisfaction in a dynamic and competitive environment.

Keywords: Churn, DNN, dataset, epochs, prediction, optimization.

1. Introduction
Telecommunication companies find it more economical advantageous in maintaining and keeping existing customers rather than loose them and bringing in new ones. The majority of telecom firms view their customers as their most valuable asset. For many markets, maintaining good customer relations is an important yet costly endeavor. This is especially crucial for the telecom industry, as companies that have made significant infrastructure investments. Existing customers are invaluable assets due to their established relationship with the company, which translates into ongoing revenue and lower marketing costs. Moreover, retaining customers can enhance brand loyalty, improve customer lifetime value, and boost overall profitability (Kumar & Shah, 2004) [10]. Conversely, losing existing customers can have substantial disadvantages. High churn rates can lead to significant revenue losses, increased acquisition costs for new customers, and a potential decline in market share. Additionally, customer attrition can negatively impact brand reputation and disrupt the stability of the customer base (Reinartz & Kumar, 2002) [12].

To mitigate these risks, telecom companies have historically employed various approaches to predict customer attrition. Early methods included statistical techniques such as logistic regression and decision trees, which provided valuable insights but often struggled with complex, non-linear relationships within the data (Berry & Linoff, 2004) [5]. More recently, advancements in machine learning have introduced sophisticated techniques like support vector machines (SVMs) and ensemble methods, which offer improved accuracy and predictive power (Friedman 2001) [8].

In order to forecast customer attrition, deep neural networks (DNNs) have become extremely effective tools in this field. DNNs can process big and complicated information. Using their capacity to recognize complex links and patterns, DNNs can offer telecom businesses insightful information that lowers churn rates (Adwan *et al.*, 2014; Vural *et al.*, 2020) [2, 16]. The use of feature engineering in DNNs for churn prediction is crucial. Relevant features that telecom companies can extract include conversation duration, data consumption, billing history, and consumer demographics. These features are then fed into a neural network. Studies conducted by Vural *et al.* (2020) [16]

emphasize how crucial feature selection and preprocessing are to raising the predicted accuracy of the model. Feature sets that are well-designed improve the DNN's capacity to identify faint signals that point to possible churn. When it comes to capturing temporal dependencies in client behavior, DNNs are especially useful. They are able to determine consumption trends and changes over time by analyzing a customer's historical data.

The scalability of DNNs is another important benefit. With DNNs, telecom businesses can effectively manage this big data dilemma as they gather enormous amounts of client data. Employing designs such as convolutional neural networks (CNNs), deep feedforward neural networks, or more sophisticated models like deep autoencoders, businesses are able to sort through massive information and identify minute customer behaviors that point to churn risk. Researchers using deep learning frameworks and distributed computing platforms, such (Almufadi *et al.* 2019) [3] have demonstrated the scalability of DNNs for telecom churn prediction. For telecom companies, churn prediction with deep neural networks (DNNs) presents a promising way to keep customers. DNNs are a valuable tool for customer churn prediction because they can automatically learn complex patterns, handle temporal dependencies, and scale to large datasets. By optimizing feature engineering and utilizing sophisticated architectures such as CNNs and RNNs, telecom companies can leverage the predictive power of deep learning to improve customer retention and lower churn rates.

A major problem in the telecom sector is customer attrition, or the loss of subscribers. Telecom providers must correctly identify the consumers at risk of leaving in order to solve this issue. While traditional churn prediction models have provided valuable insights, there is an increasing demand for more accurate, data-driven solutions. Deep Neural Networks (DNNs) offer a promising approach due to their ability to handle large volumes of heterogeneous customer data and identify complex patterns that traditional methods might overlook. However, the application of DNNs to churn prediction in telecom is not without challenges. To effectively harness DNNs, telecom companies must design, train, and deploy models capable of efficiently processing massive, diverse, and time-dependent customer data while maintaining interpretability and scalability. Addressing these challenges is crucial for developing accurate and actionable churn prediction systems that can significantly reduce customer churn and enhance customer retention strategies.

2. Review of related works

Baby *et al.* (2023) [4] examined customer churn prediction in the banking sector using Artificial Neural Networks (ANNs). The methodology involved training the ANN model with input features to predict churn as the independent variable. The model was developed by adjusting hyperparameters, utilizing the forward propagation algorithm, and applying cross-validation techniques. Forward propagation involves passing input data through the network layers, calculating outputs, and updating model parameters accordingly. Hyperparameter tuning included optimizing aspects like learning rate, hidden layers, and neuron counts, while cross-validation ensured the model's robustness and ability to generalize. The ANN model achieved an accuracy of 86% in predicting churn, outperforming a logistic regression model and demonstrating its effectiveness in capturing complex data relationships. This high accuracy underscores the ANN model's superior performance in identifying potential churners. The study highlights the practical value of this model for the banking industry, where it can be used to proactively identify at-risk customers and implement effective retention strategies.

Sikri *et al.* (2024) [14] tackled customer churn prediction using machine learning, addressing challenges related to imbalanced and diverse data distributions in churn datasets. The study introduced a novel Ratio-based data balancing technique during preprocessing to mitigate data skewness, a common issue leading to biased models. The research evaluated several algorithms, including Perceptron, Multi-Layer Perceptron, Naive Bayes, Logistic Regression, K-Nearest Neighbour, Decision Tree, and ensemble methods like Gradient Boosting and XGBoost. These algorithms were tested on datasets balanced using the proposed technique and traditional resampling methods like Over-Sampling and Under-Sampling. Results showed that the Ratio-based technique significantly outperformed traditional methods, enhancing churn prediction accuracy. Ensemble algorithms, particularly Gradient Boosting and XGBoost, delivered superior results compared to standalone methods, with the best outcomes achieved using a 75:25 data ratio and XGBoost. The study highlighted the effectiveness of this approach in improving model performance, as measured by Accuracy, Precision, Recall, and F-Score.

Oladipo *et al.* (2023) [11] focuses on predicting customer churn in the telecommunications industry using an ensemble-based approach. The model integrates several machine learning algorithms, including XGBoost, LightGBM, Random Forest (RF), and CatBoost. These algorithms are combined using a stacking technique, which involves training multiple models and then combining their predictions to improve overall performance. Metaheuristics are also developed to enhance the predictive capabilities of the ensemble model. The model leverages ensemble learning, a technique that improves prediction accuracy by combining the strengths of various machine learning algorithms. Stacking is used to integrate XGBoost, LightGBM, RF, and CatBoost, each contributing unique strengths to the predictive model. This approach helps the telecommunications company predict customer attrition by analyzing patterns in customer data and identifying those most likely to churn. The use of metaheuristics further refines the model, optimizing its performance. The result showed that the ensemble-based model achieved an impressive accuracy of 92.2% in predicting customer churn. This high level of accuracy demonstrates the effectiveness of combining multiple algorithms through stacking and metaheuristics in forecasting customer attrition. The results highlight the model's capability to accurately identify at-risk customers, providing valuable insights for the telecommunications industry to enhance customer retention strategies.

Suh, Y (2023) [15] presented the application of a machine learning algorithm to predict customer churn in the rental business sector, specifically for a water purifier rental company. The algorithm was trained on a large dataset to learn meaningful features that contribute to churn. Performance metrics such as the F-measure and Area under Curve (AUC) were used to evaluate the model's

effectiveness. The model was developed by analyzing customer behavior data from a Korean electronics company's rental service. The dataset used for training and testing contained approximately 84,000 customers, providing a substantial basis for the algorithm to learn from. The model's performance was further validated using a larger dataset of about 250,000 customers to test its predictive capabilities in real-world scenarios. The model not only predicted churn but also identified key variables influencing churn, which could be used by customer management staff to implement targeted marketing strategies. The model achieved impressive results, with an F1 score of 93% and an AUC of 88%, indicating strong predictive accuracy and reliability. Additionally, the model's inference performance on a larger operational dataset demonstrated a hit rate of about 80%, confirming its effectiveness in predicting actual churn cases. The identification of influential variables on churn provided actionable insights for personalized marketing efforts, helping the company address churn causes more effectively.

Dhangar & Anand (2021) [7] reported a comprehensive methodology for predicting customer churn across various industries. The approach begins with data pre-processing and exploratory data analysis, followed by feature selection. The dataset is then split into training (80%) and testing (20%) sets. Multiple machine learning algorithms, including Logistic Regression, Naive Bayes, Support Vector Machine (SVM), and Random Forest, are applied to the training data. Ensemble techniques are also utilized to assess their impact on model accuracy. K-fold cross-validation is employed for hyperparameter tuning and to prevent overfitting. Finally, the models' performance is evaluated using the AUC/ROC curve. The results indicate that Random Forest achieved the highest accuracy (87%) and the best AUC score (94.5%), making it the most effective model for predicting customer churn in this study. The SVM classifier also performed well, with an accuracy of 84% and an AUC score of 92.1%. These results suggest that Random Forest, followed closely by SVM, outperforms other models in predicting customer churn.

Abou-el-Kassem *et al.* (2020) [1] presented a dual-approach methodology to address customer churn in the telecom sector. The first approach involves identifying key factors influencing customer churn using machine learning algorithms such as Deep Learning, Logistic Regression, and Naive Bayes. A dataset is built from practical questionnaires to facilitate this analysis. The second approach focuses on predicting customer churn by analyzing user-generated content (UGC) from social media platforms. Sentiment analysis is applied to this UGC to determine text polarity, categorizing it as positive or negative. The results indicate that the machine learning algorithms employed—Deep Learning, Logistic Regression, and Naive Bayes—achieved similar accuracy levels in predicting customer churn. However, the algorithms differed in how they weighted the importance of various attributes in the decision-making process. This suggests that while the models are equally effective in accuracy, they prioritize different factors when making predictions.

Hamuntenya & Iyawa (2023) [9] reported the application of four machine learning algorithms (K-Nearest Neighbors, Random Forest, Gradient Boosting Tree, and XGBoost) to predict customer churn for MTC Namibia, a mobile network operator. The use of these algorithms reflects a common approach in churn prediction, where machine learning techniques are employed to analyze customer behavior and predict potential attrition. The model focused on key factors such as real monthly usage, plan choice, and payment methods. These features were used as inputs to train the models. The performance of the models was evaluated using the ROC-AUC (Receiver Operating Characteristic - Area under the Curve) score, a metric that measures the model's ability to distinguish between classes (churn and no churn). Among the algorithms tested, XGBoost emerged as the top-performing model, achieving an accuracy rating of 84% based on the ROC-AUC score. The study also identified that specific factors—real monthly usage, plan choice, and payment method—significantly influence churn rates, highlighting the importance of these variables in predicting customer attrition.

Zhao (2023) [17] focused on predicting customer churn in a banking context using two machine learning algorithms: Random Forest and Decision Tree classifiers. The methodology begins with data preprocessing, which involves cleaning and adjusting the dataset by removing irrelevant features and renaming feature names for better accessibility during analysis. The dataset is then split into training and testing sets using an 80-20 split. The study proceeds by building churn prediction models with the selected algorithms. Feature selection is performed to identify the most significant variables, and their importance is visualized using bar graphs. Both models are trained on the training set and subsequently tested on the testing set. The performance of the models is evaluated using confusion matrices and accuracy scores, which are standard tools for assessing classification models. The visualization of these metrics allows for a clear comparison of model effectiveness. The results indicate that both the Random Forest and Decision Tree models successfully predict customer churn, with the Random Forest model outperforming the Decision Tree model. The Random Forest model achieved a higher accuracy score of 91%, making it the more effective algorithm in this study.

Çalış & Kozłowska (2021) [6] focused on predicting customer churn using various data mining techniques and classification algorithms within a machine learning framework. The methodology involves analyzing a dataset of 7,166 customer records from a telecommunications company. The study aims to identify the most effective algorithms: logistic regression, K-Nearest neighbor, Decision tree, random forest, Support Vector Classifier for accurately predicting customer churn. The process begins with data preprocessing, which includes scaling and applying log transformations to the data, likely to normalize the features and enhance model performance. Classification algorithms are employed to build predictive models, although the abstract does not specify which algorithms were used. The results show that logistic regression was the top performing model with an accuracy of 81% and also had the best recall and decision tree was closely following with an accuracy of 57%. Their work indicated that the model struggled with an unbalanced data which affected the performance of the models. According to other metric, the best model is the decision tree because it produced a better recall outcome.

Shrestha & Shakya (2019) [13] addressed a critical issue in the telecommunications industry: customer churn, which significantly impacts revenue. The study emphasizes the

importance of effective churn management for gaining a competitive edge by improving customer retention rates. The research highlights the challenge of handling imbalanced datasets in churn prediction, particularly with real-world telecommunication data that may differ from publicly available datasets. The methodology involves applying the XGBoost machine learning algorithm to both a native dataset from a major telecommunications company in Nepal and a publicly available dataset. The native dataset comprises 52,332 customer records, with a significant imbalance between non-churned (46,204) and churned (6,128) customers. The performance of XGBoost on this dataset is impressive, yielding an accuracy of 97% and an F1-score of 88%. Additionally, the study compares results with a publicly available dataset of 3,333 subscribers, achieving slightly lower but still strong accuracy (96.25%) and F1-score (86.34%). The high accuracy and F1-scores underscore the algorithm's effectiveness in predicting customer churn, offering valuable insights for telecom companies aiming to improve retention strategies.

3. Methodology: The proposed customer attrition model in a telecom company is a machine learning approach to solving churn problems. It involves the evaluation of the performance and effectiveness of DNN model in customer attrition. The implementation of a Deep Neural Network (DNN) for forecasting customer attrition involves a systematic approach to building a robust predictive model capable of accurately identifying customers at risk of churning. The process begins with the acquisition and preprocessing of telecom customer data, including handling missing values, and normalizing features. Feature engineering and selection are then performed to identify the most relevant attributes that influence customer attrition. The DNN model is constructed with multiple layers to capture complex patterns within the data. Hyperparameters, including learning rate, batch size, and the number of hidden layers, are fine-tuned through methods such as cross-validation to optimize the model's performance. The trained model is then evaluated using metrics like accuracy, precision, recall, and F1-Score, demonstrating its effectiveness compared to traditional machine learning methods. The successful deployment of this DNN model enables the telecom firm to proactively identify and retain at-risk customers, thereby improving customer satisfaction and reducing churn-related revenue losses.

The system's architectural framework shown in figure 1 involves the design and organization of the components, layers and processes that is involved in the prediction system. It guides the development and implementation of the system and ensures that the system is efficient, scalable and capable of providing actionable insights for customer retention

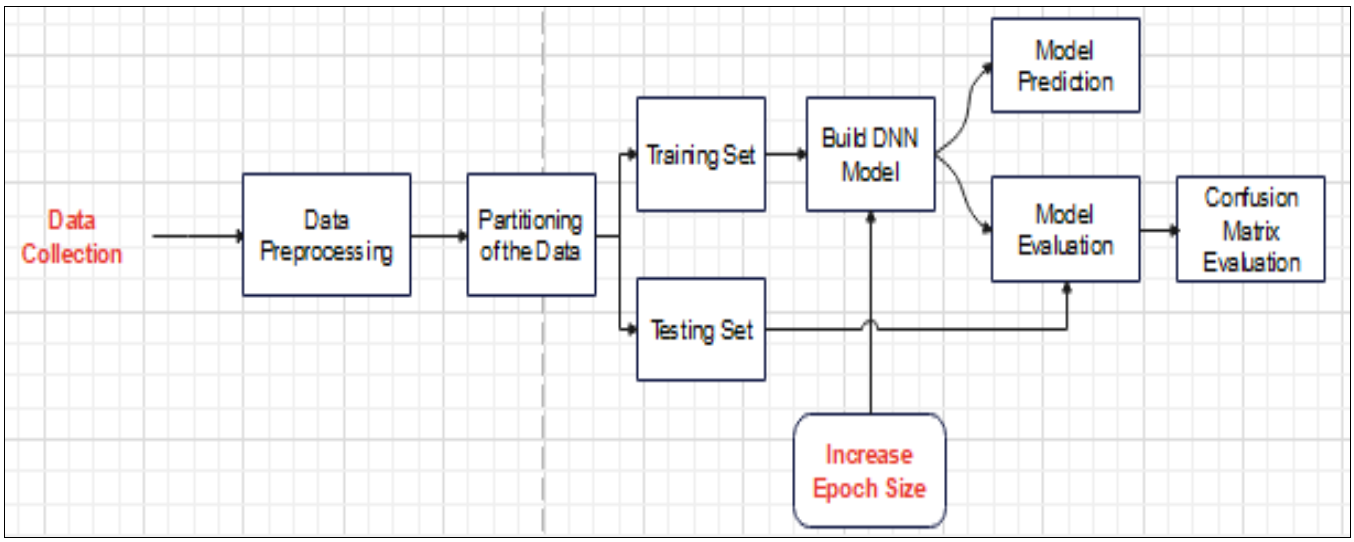

Fig 1: Architectural Framework of the Customer attrition for a Telecom Company

Model Development

DNN Model Development: In developing the DNN model, we selected a DNN architecture that is suited for the challenge according on its complexity. The deep feedforward networks with several hidden layers was adopted. The model's input features were Specified, these are the churn-related features and pertinent customer data that were prepared during the data pretreatment stage. To ascertain the total number of neurons in each layer as well as the number of hidden layers. several architectures were examined to determine the one that works best for the dataset, the right activation functions was selected for the hidden layer neurons, which is the Sigmoid functions which mapped input to value between 0 and 1 which is suitable for binary classification problems, σ (x)=11+e−x\sigma(x) = \frac{1}{1 + e^{-x}}σ(x)=1+e−x1 and ReLUs (Rectified Linear Units) ReLU(x)=max(0,x) was also selected, as it mapped all negative values to 0. An output layer was created that generates a customer attrition. The developed deep neural network architecture is shown in Figure 2

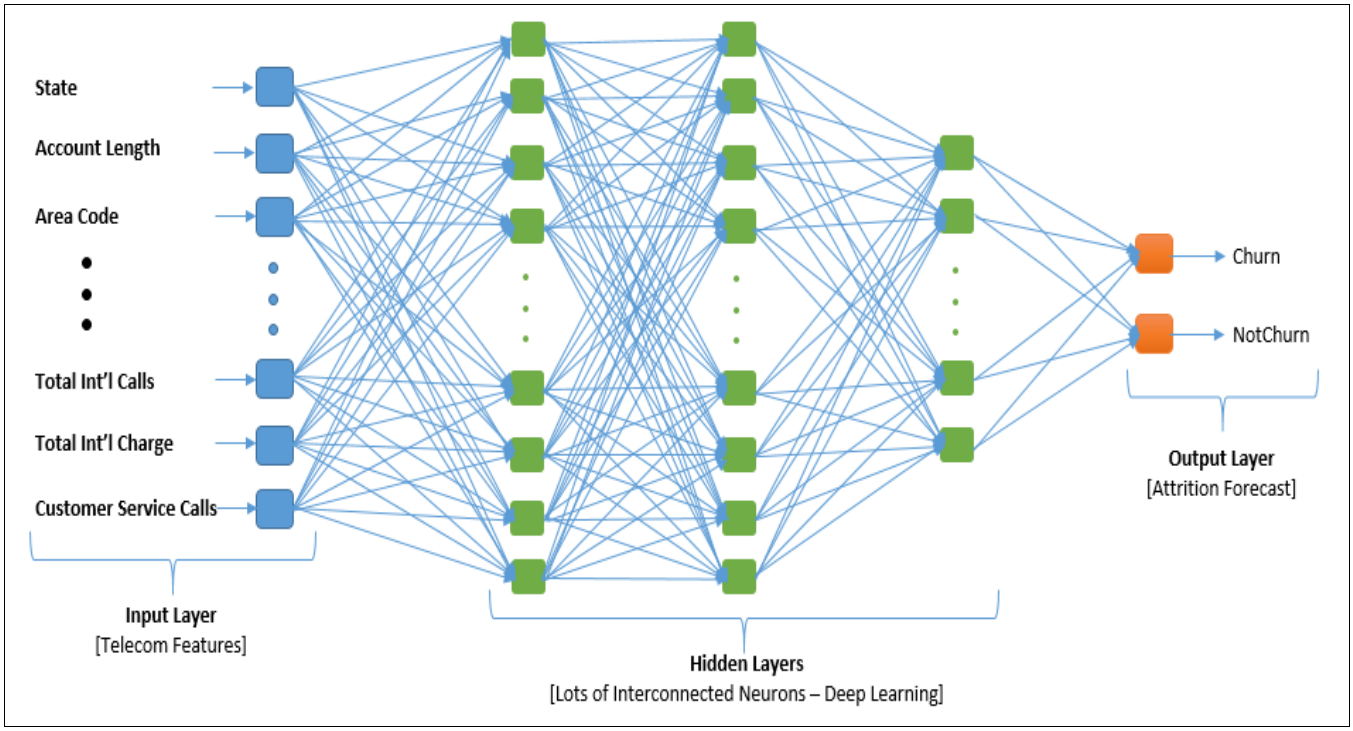

Fig 2: Developed Deep Neural Network Architecture

Loading of Dataset Code Snippet

The dataset is loaded as input into the DNN architecture. The already downloaded file is stored in a folder in the system where the code file is stored and is easily loaded to the system for manipulation by the model.

```
# Load telecom churn dataset
# Example dataset can be found here: https://www.kaggle.com/blastchar/telco-customer-churn
telecom_data <- read.csv("churn-bigml-80.csv")
test_data <- read.csv("churn-bigml-20.csv")
```

Code Snippet for Model Development

The code snippet shown above is the developed DNN model/ architecture used for the training of the dataset to learn the patterns in the dataset to be able to forecast attrition with new unseen data. The model uses the ReLu activation function in the hidden layers and the sigmoid activation function in the output layer of the model.

Model Training

The size of the dataset is 3,333 instances and 80% was used to train the model while 20% was used for testing.

We selected a loss function that measures the discrepancy between the actual churn labels and the model's predictions, the binary cross-entropy technique was used. The Binary Cross-Entropy also called Log Loss is the standard loss function for binary classification problems and It is suitable for this task, as it works well with models that output probabilities, it also measures how far the predicted probabilities are from the actual binary labels (0 or 1).

Model Optimization

To enhance the model's performance and ensure its ability to generalize well to unseen data, a combination of advanced model optimization techniques, including hyper-parameter tuning, regularization, and early stopping where chosen. Each of these methods played a critical role in improving the accuracy, stability, and generalization ability of the model.

Hyper-parameter Tuning

Essential hyper-parameters such as the number of neurons in each layer were optimized, the number of hidden layers, batch size, and learning rate. To systematically and efficiently search for the optimal set of hyper-parameters, Bayesian optimization was adopted. This technique constructs a probabilistic model of the objective function and uses it to select the most promising hyper-parameters to evaluate next. Compared to grid or random search, Bayesian optimization reduces the number of trials required to reach near-optimal performance.

To address the issue of overfitting and enhance the model's generalization capability, we applied regularization techniques, namely: *L1 Regularization (Lasso),* by Introducing a penalty equal to the absolute value of the magnitude of coefficients, encouraging sparsity in the model; *L2 Regularization (Ridge)* by Adding a penalty proportional to the square of the magnitude of coefficients, discouraging large weight values; and *Dropout Layers*, where a random subset of neurons is deactivated in each training iteration, thereby reducing reliance on specific features and improving robustness. These regularization strategies helped control model complexity and prevent it from memorizing the training data. When validation loss began to increase, indicating that the model was starting to over fit the training data, these regularization techniques ensured that the final model retained strong predictive power without unnecessary training iterations

Feature Scaling

The application of feature scaling to the dataset helped to prevent feature dominance, where features with large ranges dominate the model.

The steps adopted to apply feature Scaling are;

1. Importation of necessary libraries: from sklearn.preprocessing import StandardScaler
2. Creation of a scaler object: scaler = StandardScaler()
3. Fit the scaler object to the data: scaler.fit(X_train)
4. Transformed the data: X_train_scaled = scaler.transform(X_train)
5. Applied the same scaling to the test data: X_test_scaled = scaler.transform(X_test)

For Feature Selection which helped reduce dimensionality, improve model performance, and prevent over fitting

Steps followed

1. Importation of necessary libraries: from sklearn.feature_selection import SelectKBest, mutual_info_classif
2. Created a feature selector object: selector = SelectKBest(mutual_info_classif, k=10)
3. `K` determines the number of features to select.
4. Fitted the selector to your data: selector.fit(X_train, y_train)
5. This step calculates the mutual information between each feature and the target variable.
6. Transform your data: X_train_selected = selector.transform(X_train)
7. This step applies the feature selection to your training data.
8. We applied the same selection to the test data: X_test_selected = selector.transform(X_test) this ensures that the same features are selected for both training and testing.

For handling imbalanced datasets

The SMOTE (Synthetic Minority Over-Sampling Technique) Techniques helped address class imbalance issues in attrition datasets.

The SMOTE steps followed are

Importation of libraries: from imblearn.over_sampling import SMOTE

1. Creation of a SMOTE object: smote = SMOTE(random_state=42)
2. random_state ensures reproducibility of the results.
3. Fitted the SMOTE object to the data: X_train_resampled, y_train_resampled = smote.fit_resample(X_train, y_train)

These step generated synthetic samples of the minority class and were combined with the original data.

These formulas provide a mathematical foundation for understanding how Feature Scaling, Feature Selection, and SMOTE work

- Standardization (Z-Score Normalization): `z = (x - μ) / σ`
- `x`: original feature value
- `μ`: mean of the feature
- `σ`: standard deviation of the feature
- `z`: scaled feature value

Min-Max Scaling: `x_scaled = (x - x_min) / (x_max - x_min)`

- `x`: original feature value
- `x_min`: minimum value of the feature
- `x_max`: maximum value of the feature
- `x_scaled`: scaled feature value

Feature Selection

Mutual Information: `I(X;Y) = ∑∑ p(x,y) log(p(x,y)/(p(x)p(y)))`

- `X`: feature
- Y`: target variable
- `p(x,y)`: joint probability distribution of `X` and `Y`
- `p(x)`: marginal probability distribution of `X`
- `p(y)`: marginal probability distribution of `Y`
- `I(X;Y)`: mutual information between `X` and `Y`

Smote

- Synthetic Sample generation `x_synthetic = x_i + (x_zi - x_i) * δ`
- `x_i`: minority class sample
- `x_zi`: another minority class sample (randomly selected)
- `δ`: random number between 0 and 1
- `x_synthetic`: synthetic sample generated

4. Results

The exploratory data analysis (EDA) is used to understand the distribution of data and variable relationships. From the results obtained, key patterns and features associated with customer attrition that facilitate informed decision-making process for the telecom firm are shown in table 1. The model validation graphs for 10 and 100 epochs are presented in figure 3 and figure 4 respectively, while table 2 and table 3 shows their respective confusion matrices. The statistical result summary is presented in table 4.

Table 1: Feature Importance Score

Features	Mean Decrease Gini
State	98.903401
Account Length	17.928327
Area Code	4.709509
International Plan	52.726265
Voice Mail Plan	8.992156
Number of Vmail Messages	14.876575
Total Day Minutes	76.118072
Total Day Calls	17.505372
Total Day Charge	78.532000
Total Eve Minutes	33.655187

Total Eve Calls	15.715358
Total Eve Charge	34.138874
Total Night Minutes	20.181908
Total Night Calls	17.074633
Total Night Charge	19.820026
Total Intl Minutes	25.868424
Total Intl Calls	29.046591
Total Intl Charge	24.009036
Customer Service Calls	73.435423

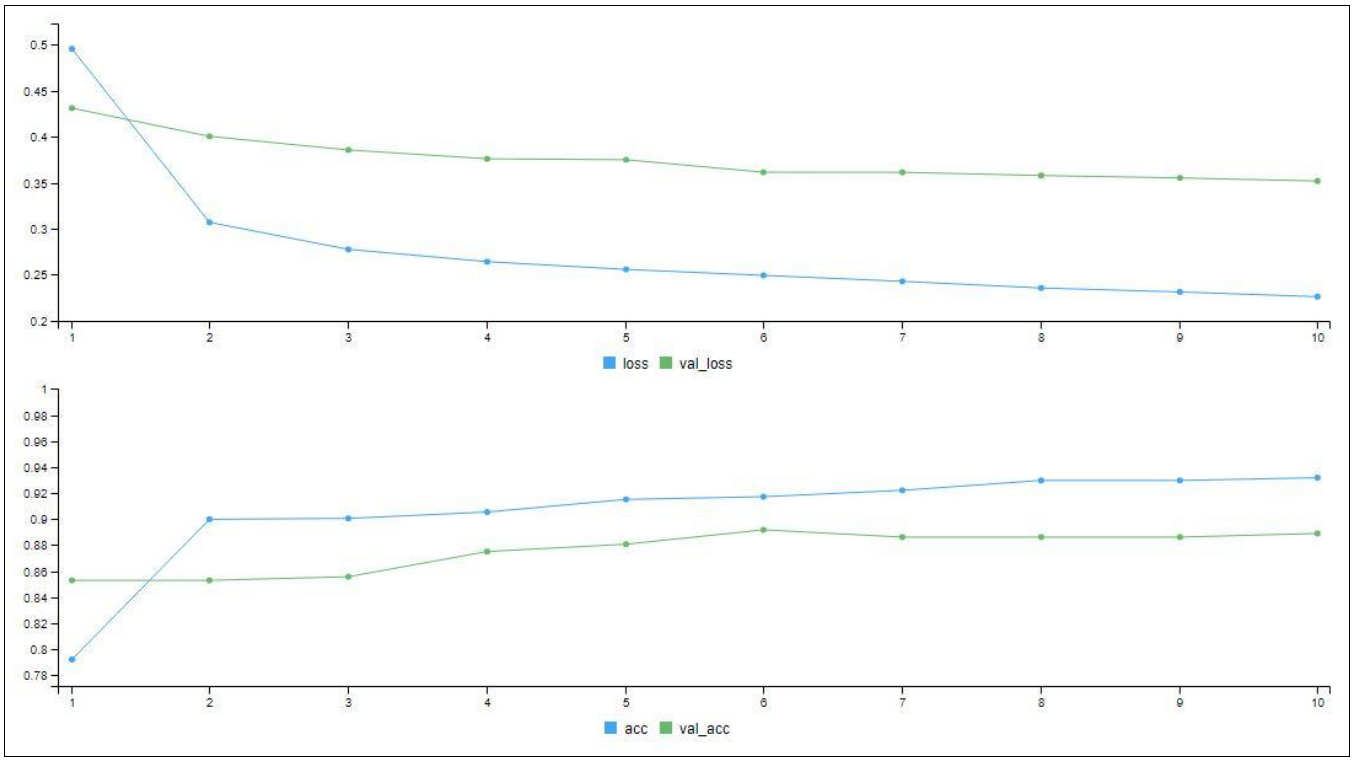

Fig 3: Model Validation Graph for 10 Epoch (Iteration)

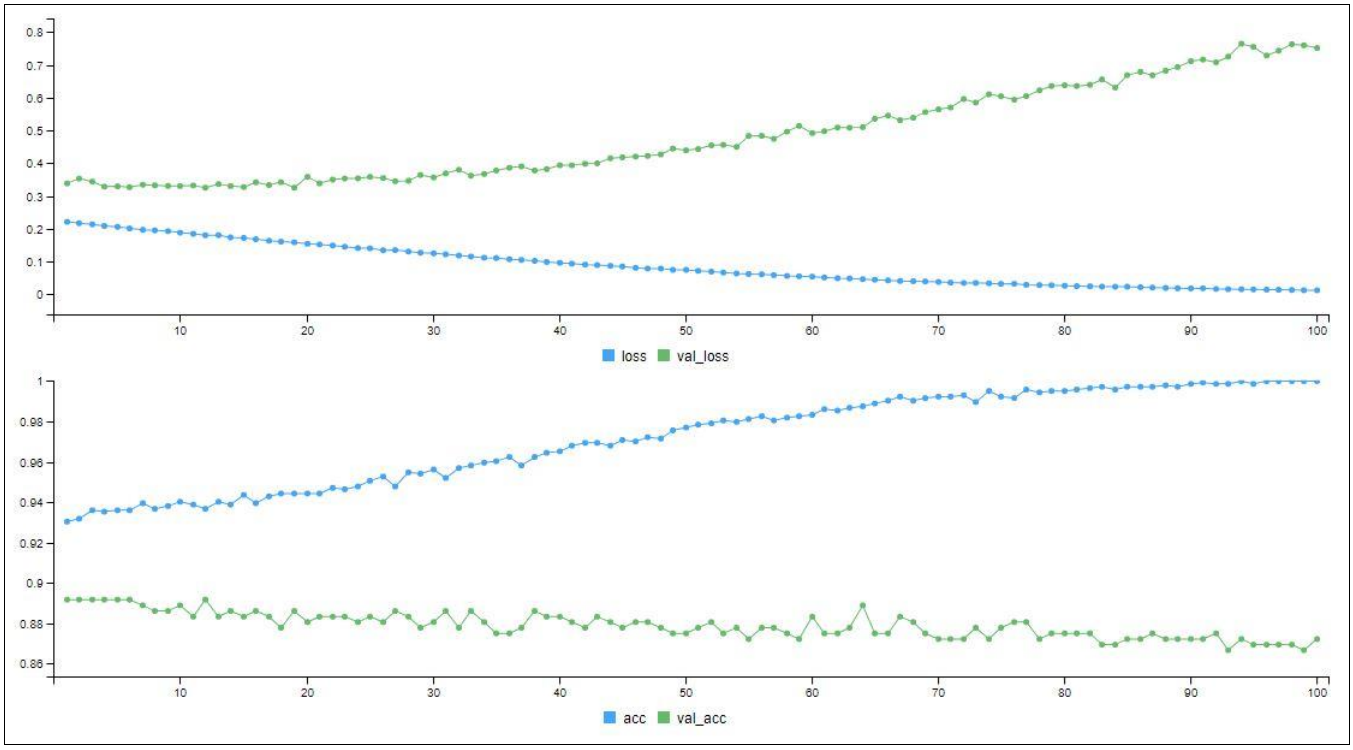

Fig 4: Model Validation Graph for 100 Epoch (Iteration)

Table 2: Confusion Matrix for 10 Epoch (Iterations)

Prediction	Reference	
	0	**1**
0	397	36
1	3	13

Table 3: Confusion Matrix for 100 Epoch (Iterations)

Prediction	Reference	
	0	**1**
0	389	29
1	11	20

Table 4: Statistical Result Summary

	Precision	Sensitivity	Specificity	P-Value	Accuracy
10 Epoch	92%	99%	27%	2.99e-07	91%
100 Epoch	93.1%	97%	41%	0.00719	98.1%

5. Discussion of Results

The model training was done twice with 10 and 100 epochs (i.e. iterations). Figure 3 is the model validation graph for 10 iterations. It shows the loss - validation loss and accuracy - validation accuracy of the model. We can see the loss line (blue line) coming far below the validation loss line (green line). At the final iteration, the loss line came down to a little above 0.2 mark. The accuracy of the model also climbed from 78% to a little above 90% as seen in the graph. In Figure 4.6, after the 100 iterations, we can see the loss line (blue line) coming far below the validation loss line (green line). At the final iteration, the loss line came down below 0.1 mark. The accuracy (blue line) of the model also climbed from 93% to above 98% as seen in the graph.

In Figure 4. After the 100 iterations, we can see the loss line (blue line) coming far below the validation loss line (green line). At the final iteration, the loss line came down below 0.1 mark. The accuracy (blue line) of the model also climbed from 93% to above 98% as seen in the graph.

The misclassification table for 10 epoch as shown in Table 2 depicts that, from the reference (actual) prediction, 13 customers has churn intensions and the model predicted the same. The model also predicted that 397 customers has no plan of churning and the reference predicted same. The model misclassified or was confused with 36 customers predicting that they have no plan of churning while the reference predicted otherwise and the model predicted that 3 customers has churn in mind while in actual case, it is otherwise.

Table 3 is the misclassification table for 100 epochs. The model predicted 389 customers has no churn plans and the actual prediction was same. The model also predicted that, 20 customers will churn and the reference predicted same. However, the model was confused with some records in that, it predicted that 29 customers will not churn whereas in actual sense they churned as the model also misclassified 11 customers as it predicted that they will churn, but they didn't. Table 4 is the summary of the statistical results produced by the DNN model for customer attrition. From the table, we can see that there was a slight improvement when the model was trained with 100 epoch. The stochastic gradient descent (SGD) algorithm is employed for training and optimization, iteratively refining the model's parameters to enhance its predictive accuracy. Evaluation of the developed DNN model follows, utilizing common evaluation metrics such as accuracy, precision, recall, and the confusion matrix. Findings from the evaluation reveal promising results, with the model achieving a performance accuracy of 91% after 10 epochs, and a slight improvement to 98.1% after 100 epochs. Notably, there was a concurrent enhancement in precision, recording 92% at 10 epochs and a further increase to 93.1% after 100 epochs. There was reduction in sensitivity (recall), but generally, the model recorded an improvement when iterated 100 times.

6. Conclusion

The study on customer attrition forecast for the telecom firm through the implementation of a Deep Neural Network (DNN) has provided valuable insights into the effectiveness of leveraging advanced machine learning techniques for customer retention strategies. Through the collection and integration of historical data from diverse sources, the preprocessing steps ensured the dataset's readiness for training a sophisticated DNN model. The careful splitting of the data into training and validation sets facilitated robust model development, allowing for iterative adjustments to enhance predictive accuracy. The implementation of the stochastic gradient descent (SGD) algorithm further optimized the DNN model, leading to commendable performance metrics.

References

1. Abou el Kassem E, Ali S, Mostafa A, Alsheref FK. Customer churn prediction model and identifying features to increase customer retention based on user-generated content. Int J Adv Comput Sci Appl. 2020;11(5). doi:10.14569/IJACSA.2020.0110567
2. Adwan O, Faris H, Jaradat K, Harfoushi O, Ghatasheh N. Predicting customer churn in telecom industry using multilayer perceptron neural networks: Modeling and analysis. Life Sci J. 2014;11(3):75-81.
3. Almufadi N, Qamar AM, Khan RU, Othman MTB. Deep learning-based churn prediction of telecom subscribers. Int J Eng Res Technol. 2019;12(12):2743-8.
4. Baby B, Dawod Z, Sharif S, Elmedany W. Customer churn prediction model using artificial neural network: A case study in banking. In: 2023 Int Conf Innov Intell Informatics Comput Technol (3ICT). IEEE; 2023. p. 154-61. doi:10.1109/3ICT60104.2023.10391374
5. Barry MJA, Linoff GS. Data mining techniques for marketing, sales and customer relationship

management. Indianapolis: Indianapolis Publishing Inc.; 2004.
6. Çalış A, Kozłowska J. Customer churn prediction with popular machine learning algorithms [Internet]. Wydział Inżynierii Zarządzania, Politechnika Białostocka; 2021. Available from: https://open.icm.edu.pl/items/607a4b12-7550-4fd8-9c9d-6537993f6294
7. Dhangar K, Anand P. A review on customer churn prediction using machine learning approach. Int J Innov Eng Res Technol. 2021;8(5). doi:10.17605/OSF.IO/ACNKJ
8. Friedman JH. Greedy function approximation: A gradient boosting machine. Ann Stat. 2001;29(5):1189-232. doi:10.1214/aos/1013203451
9. Hamuntenya L, Iyawa G. Enhancing customer retention: A study on churn prediction models for MTC Namibia using machine learning algorithms. In: Int Conf Inf Syst Emerg Technol (ICISET). 2023. doi:10.2139/ssrn.4642909
10. Kumar V, Shah D. Building and sustaining profitable customer loyalty for the 21st century. J Retail. 2004;80(4):317-329. doi:10.1016/j.jretai.2004.10.007
11. Oladipo ID, Awotunde JB, AbdulRaheem M, Taofeek Ibrahim FA, Obaje O, Ndunagu JN. Customer churn prediction in telecommunications using ensemble technique. Univ Ibadan J Sci Log ICT Res. 2023;9(1). doi:10.48084/uijslictr.1141
12. Reinartz WJ, Kumar V. The mismanagement of customer loyalty. Harv Bus Rev. 2002;80(7):86-94.
13. Shrestha SM, Shakya A. A customer churn prediction model using XGBoost for the telecommunication industry in Nepal. Procedia Comput Sci. 2022;215:652-661. doi:10.1016/j.procs.2022.12.067
14. Sikri A, Jameel R, Idrees SM, Kaur H. Enhancing customer retention in telecom industry with machine learning driven churn prediction. Sci Rep. 2024;14(1):13097. doi:10.1038/s41598-024-63750-0
15. Suh Y. Machine learning based customer churn prediction in home appliance rental business. J Big Data. 2023;10:41. doi:10.1186/s40537-023-00721-8
16. Vural U, Okay ME, Yildiz EM. Churn prediction for telecommunication industry using artificial neural networks. Int J Comput Inf Eng. 2020;14(11):396-399.
17. Zhao S. Customer churn prediction based on the decision tree and random forest model. BCP Bus Manag. 2023;44:339-344. doi:10.54691/bcpbm.v44i.4840