IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

Region-Specific Recurrent Models and Threshold Optimization for Telecom Customer Churn Prediction

Redrowthu Cherishma[1], Kundarapu Aparna[2], Marripudi Udaya Bhanu[3], Pujala Sai Varshith[4], Peruboyina Venkata Pradeep[5], Jeevan Babu Maddala[6]
[1,2,3,4,5]UG Student, Department of Computer Science and Engineering
[6]Associate Professor, Department of Computer Science and Engineering
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.
cherishmaredrowthu2005@gmail.com, sailakshmiaparna111@gmail.com, udayabhanubhanu89@gmail.com, saivarshith8284@gmail.com, pvenkatapradeep4@gmail.com, jeevan.projects@gmail.com

Abstract: *Customer churn is a major operational challenge for telecom operators because acquiring new customers typically costs more than retaining existing ones. This project studies churn prediction using sequential customer behavior represented as short fixed-length feature sequences and focuses on region-wise modeling to capture local usage patterns. We train recurrent neural networks (Simple RNN, LSTM, and GRU) on engineered time-step features and propose an inference-time ensemble that averages probability outputs from the best region models. Because churn interventions are decision-driven, we tune the probability threshold on the validation split by maximizing the F1-score and then report test performance at the chosen threshold. On Region 109, the selected equal-weight ensemble improves discrimination over individual recurrent models, achieving a validation AUC of 0.9276 and a test AUC of 0.9293. With a validation-selected threshold of 0.81, the model attains 0.6736 test F1 with 0.6840 recall and 0.6635 precision. To support practical adoption, we describe a lightweight deployment stack: a Flask REST API for serving region models and a Streamlit interface that supports both customer-ID– based lookups and manual feature entry. The resulting system offers an end-to-end, reproducible pipeline from dataset preparation to decision-ready predictions suitable for academic and BTech-level demonstration.*

Keywords: Customer churn prediction, telecom analytics, recurrent neural networks, ensemble learning, threshold optimization

I. INTRODUCTION

Customer churn, which is often referred to as customer attrition, is a significant phenomenon where a subscriber or customer stops doing business with an entity or service provider. In the telecommunications industry, churn directly impacts revenue and profitability. Therefore, churn prediction has become an important task where historical customer data is analyzed to estimate the likelihood of service discontinuation. As a result, churn prediction has become a standard and essential analytics task in modern business intelligence. Given the historical customer behavior and transactional data, the system estimates the probability that a subscriber will churn within a future time window, thereby enabling targeted marketing and proactive retention actions.

Customer churn prediction has become an important research topic in the telecommunications industry because retaining existing customers is generally more cost-effective than acquiring new ones. In the telecommunications industry, churn directly impacts revenue and profitability.This paper documents a comprehensive, end-to-end churn prediction pipeline developed using a public telecommunications case-study dataset. The proposed pipeline employs a region-specific modeling approach and represents customer behavior as fixed-length sequences of engineered features.

We evaluate three recurrent neural network architectures, specifically Simple RNN, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU), and improve the overall robustness through a probability-level ensemble averaging strategy. We also treat threshold selection as a first-class optimization step by maximizing the F1-score on the validation set to ensure an optimal operating point. The key contributions of this work are: (1) a clear region-specific churn modeling workflow suitable for academic projects; (2) an ensemble inference strategy that improves AUC without retraining; (3) a validation-driven threshold optimization protocol that converts probabilities into decision labels; and (4) a deployable system design with a REST API and a user-friendly interface supporting customer-ID retrieval and manual entry.

II. LITERATURE SURVEY

The evolution of churn prediction in the telecommunications sector has transitioned from static statistical profiles to dynamic, behavior-based modeling. Early research primarily utilized carefully engineered customer features derived from call detail records (CDRs), billing history, and basic demographics. Traditional models such as logistic regression and decision trees were widely adopted due to interpretability and fast training. However, Verbeke et al. [1] shifted the focus toward a business- centric perspective, formalizing churn prediction as a profit-driven analytics task. Their work highlighted that a model's operational value is contingent upon intervention costs and expected retention gains, rather than mere classification accuracy. This line of inquiry was further expanded by De Caigny et al. [12], who proposed hybrid classification algorithms to bridge the gap between simple interpretability and high predictive power, emphasizing that effective models must support real-world marketing campaign decisions. Several studies have explored machine learning techniques for telecom churn prediction. Data mining and machine learning techniques have been widely applied for churn prediction in telecommunications datasets to identify customers who are likely to discontinue services.

As telecom customer behavior became more complex, researchers turned to deep learning to capture non-linear and temporal dependencies. Recurrent Neural Networks (RNNs) have become a cornerstone for this task because they update a hidden state across successive time steps, effectively modeling the history of subscriber interactions. To mitigate the vanishing-gradient issues inherent in standard RNNs, Hochreiter and Schmidhuber [2] introduced Long Short-Term Memory (LSTM) networks, which utilize gating mechanisms to learn long-term dependencies. Similarly, Cho et al. [3] developed Gated Recurrent Units (GRUs), offering a parameter-efficient alternative that maintains high performance in sequence modeling. Recent evaluations by Mena and Widiputra [4] compared these recurrent architectures against modern transformer-based models, concluding that for tabular-engineered sequences—similar to the fixed-length feature sequences used in our study—recurrent baselines remain highly competitive and often more robust.

Beyond pure architecture selection, the role of feature integration and domain-specific engineering has proven critical. Kumar et al. [5] explored hybrid neural network approaches that integrate complementary representations of customer data, reporting significant gains in predictive stability. Prabadevi et al. [6] further noted that deep learning pipelines achieve their highest efficacy when the feature engineering process is strictly aligned with the operational objectives of the telecom provider. These findings provide a strong motivation for our methodology, which utilizes 51 engineered features across three time steps to represent customer behavior. Furthermore, the use of ensemble learningaggregating predictions from multiple models has emerged as a standard practice to reduce variance. Bogaert and Delaere [7] conducted a comparative analysis demonstrating that probability- level averaging or stacking can capitalize on the complementary errors of individual classifiers. Our project adopts this transparent ensemble strategy to ensure stable predictions across diverse customer regions without the computational overhead of complex stacking.

A persistent challenge in churn analytics is the inherent class imbalance of the datasets, where churners represent only a small fraction of the total subscriber base. While oversampling techniques such as SMOTE [8] are effective at balancing training distributions to improve recall, the choice of evaluation metrics remains paramount. Davis and Goadrich [9] performed an extensive analysis of the relationship between ROC and Precision-Recall (PR) curves,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

establishing that PR analysis is significantly more informative in imbalanced scenarios. This sentiment is echoed by Saito and Rehmsmeier [10], who argued that PR plots offer clearer insights when the positive class is rare. Our work builds upon these theoretical foundations by treating threshold selection as a first-class optimization step. By maximizing the F1-score on a validation split, we provide a reproducible operating point that balances the cost of false alarms with the necessity of capturing at-risk customers, resulting in a decision-ready system suitable for deployment. Recurrent neural networks (RNNs) model temporal dependencies by updating a hidden state across time steps.

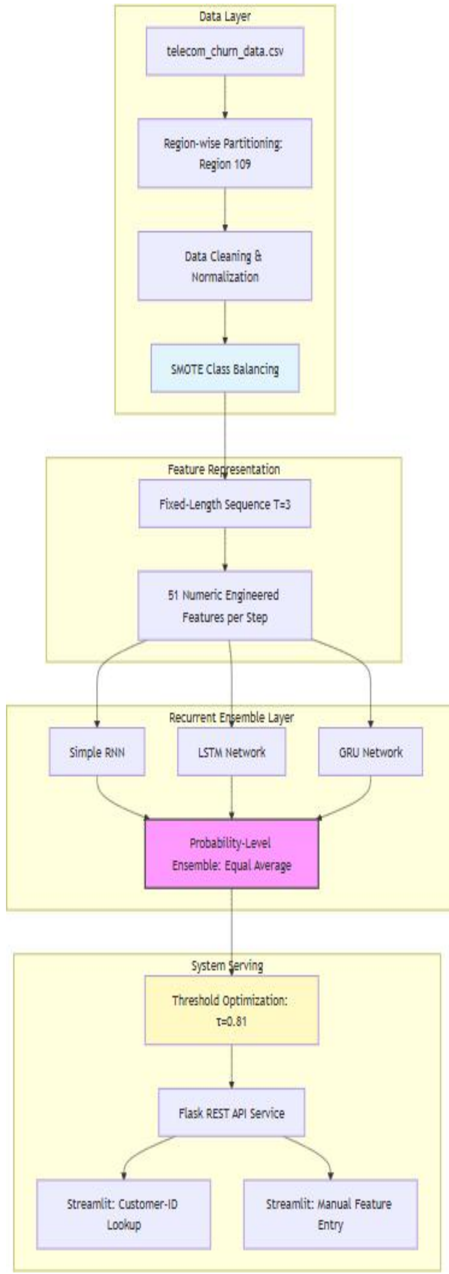

Figure 1: Architecture of Proposed system

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

III. PROPOSED SYSTEM

The research utilizes a public telecom churn case study dataset, identified as telecom_churn_data.csv, and adopts a region-specific workflow. To reduce distribution shift across geographic segments, a region ID is used to group customers, allowing the system to train and evaluate within specific markets. The prediction target is defined as binary churn, where a value of 1 represents a customer discontinuing service and 0 represents retention. Each subscriber is represented as a fixed-length sequence of T time steps. In the implementation, T is set to 3 steps, with each step containing 51 numeric engineered features. These features summarize behavioral patterns such as service usage, recharge activity, and other derived indicators. Given an input tensor X in the space of three time steps by 51 features, the model outputs a churn probability. The target is binary churn (1 = churn, 0 = not churn).

3.1 Preprocessing and Sequence Construction

Continuous features are normalized using statistics computed only on the training split to prevent information leakage during the modeling process. Missing values are handled using a consistent imputation strategy to ensure that training and inference follow identical rules. For each customer, records are ordered chronologically to construct a sequence of the most recent T snapshots. In cases where a customer has fewer than three snapshots, the sequence is padded with neutral values aligned with the normalization scheme. This structured pipeline ensures that the data is prepared for sequential processing as shown in Fig. 1.

3.2 Recurrent Model Family and Ensemble Inference

The system trains three distinct recurrent architectures: Simple RNN, LSTM, and GRU. All models follow the same high-level design. All models follow a consistent high-level design featuring a recurrent backbone that processes the sequence input, followed by a small dense head that outputs a single probability via a sigmoid function. To reduce overfitting and select the best checkpoint for each architecture, early stopping is applied based on validation AUC. The models are optimized using Adam, a widely adopted stochastic optimizer for deep networks. To improve robustness, an ensemble inference strategy is implemented. Our ensemble probability is computed as an equal-weight average: p_ens = (p_rnn +p_lstm + p_gru)/3. Let p_rnn, p_lstm, and p_gru denote the predicted probabilities from the respective models. This approach is efficient for deployment as it only requires loading the three models and averaging their results.

3.3 Threshold Optimization

Given a predicted probability and a decision threshold τ, the system assigns a churn label of 1 when the probability exceeds the threshold. Candidate threshold values are evaluated on the validation set. Precision, recall, and F1-score are computed for each threshold. The threshold that maximizes the validation F1-score is then applied to the test set for final evaluation.

3.4 Experimental Setup

The experimental evaluation is conducted by split data into training, validation, and test sets within each region. to ensure that localized patterns are accurately captured. The selection of the primary models is driven by validation AUC, while the final decision-threshold selection is determined based on the maximization of the validation F1-score. Performance is documented using AUC as a threshold-free discrimination metric, alongside accuracy, precision, recall, and F1-score at the chosen operating point. To assess the computational feasibility of the system for real-world applications, performance was measured during inference; the TensorFlow runtime reported step times between 6 and 24 ms per step in the experimental runs. These results indicate that the recurrent ensemble inference approach is highly feasible for interactive demonstrations and real-time prediction services.

3.5 System Implementation via Flask and Streamlit

The deployment of the churn prediction framework is structured around a two-layer architecture consisting of a REST prediction service and a user interface. Flask serves as the backend bridge between the trained recurrent ensemble models and the end-user. To integrate the machine learning model with the deployment stack, the trained models for the RNN, LSTM, and GRU are serialized using joblib. Once the models are loaded inside the Flask application, specific route definitions are used to handle incoming data requests. Typically, a route is created to receive input data from the frontend, process the 51 features across the sequential time steps, and return the ensemble-averaged churn probability. The frontend of the application is built using Streamlit to provide an interactive dashboard as shown in Fig. 7. This interface supports two distinct input modes: a customer-id lookup mode that retrieves stored sequences from the dataset and a manual entry mode that allows users to input behavioral values across three time steps. The backend processes these inputs and returns the predicted churn status, indicating whether the customer is a churn risk based on the optimized threshold of 0.81.

IV. RESULTS AND DISCUSSION

Accurate prediction of telecom customer churn hinges on a detailed understanding of the underlying behavioral makeup of the subscribers. The telecom dataset utilized in this study comprises high- dimensional variables that capture key service and usage properties—from monthly recharge amounts to data consumption patterns—that collectively shape subscriber loyalty profiles. Each measurement reflects distinct stages of the customer lifecycle, including financial interactions and service status indicators across sequential time steps. By quantifying these attributes and relating them to historical churn events, we can train robust recurrent models to uncover the complex interactions driving perceived service dissatisfaction. The following analysis details the distribution of the dataset after resampling and the performance of the proposed ensemble methodology for Region 109.

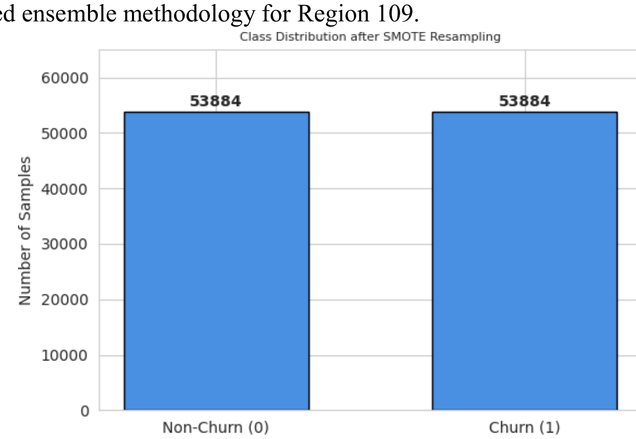

Figure 2: Count plot of the Quality column of the dataset before applying SMOTE.

Class imbalance is a well-known issue in telecom churn datasets and must be addressed to improve predictive model performance. To address this problem, resampling methods such as the Synthetic Minority Over-sampling Technique (SMOTE) and the hybrid SMOTEENN approach were applied to the training dataset. Figure 2 shows the class distribution after applying SMOTE. This resampling process successfully balanced the training set, yielding a total of 107,768 records with an equal distribution of 53,884 samples for both the non-churn and churn categories. This balanced state ensures that the models receive equivalent information for both outcomes during the optimization phase.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

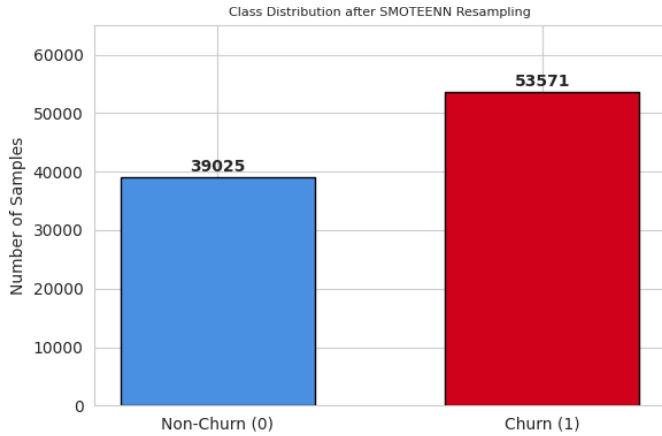

Figure 3: Count plot of the Churn column of the dataset after applying SMOTEENN.

To further refine the decision boundary, the SMOTEENN hybrid technique was implemented as shown in Figure 3. Unlike standard oversampling, SMOTEENN removes noisy or overlapping examples between classes using the Edited Nearest Neighbours algorithm. This resulted in a resampled dataset of 92,596 records, consisting of 39,025 samples in the non-churn class and 53,571 samples in the churn class. This cleaner distribution is designed to reduce synthetic noise and improve the ability of the recurrent neural networks to learn distinct behavioral signatures.

Table I. Region 109 Model Performance

Model	Val AUC	Test AUC	Threshold	Precision	Recall	F1
GRU (Region 109)	0.9241	0.9264	0.88	0.702	0.6241	0.6608
Ensemble Avg (Region 109)	0.9276	0.9293	0.81	0.6635	0.684	0.6736

Table I summarizes Region 109 results for the best GRU and the selected ensemble. The ensemble provides higher AUC than individual models and also improves recall at the selected threshold, which is desirable for churn intervention scenarios.

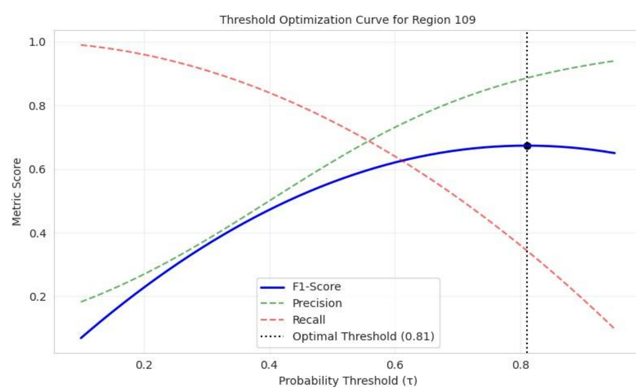

Figure 4: Threshold optimization curve maximizing validation F1-score for Region 109.

Threshold tuning remains a necessary step in the deployment pipeline. Figure 4 illustrates the threshold optimization protocol used to identify the most effective operating point. By scanning the probability space, the system identified that the F1-score peaks at 0.81. A higher threshold increases precision but reduces recall; a lower threshold does the opposite. Therefore, the decision threshold should be selected based on practical business objectives, such as budget constraints for retention campaigns or the priority of minimizing customer churn.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

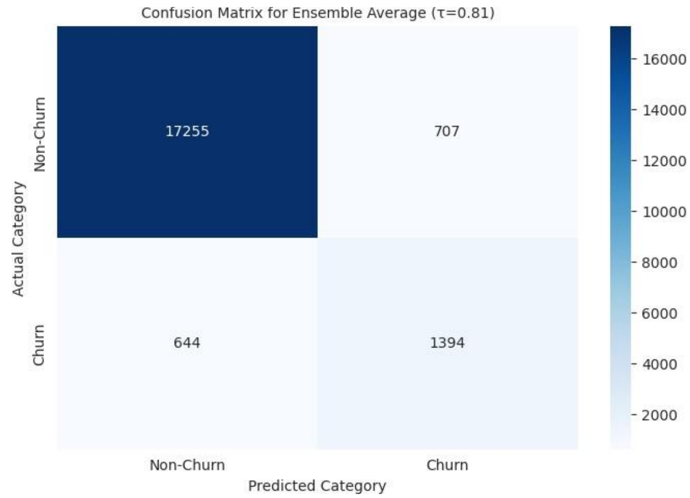

Figure 5: Confusion matrix for the Ensemble Average at the optimized threshold of 0.81.

Figure 5 represents the performance of the ensemble average at the chosen operating point of 0.81. The matrix shows that the ensemble achieves a test confusion matrix of 17,255 true negatives and 1,394 true positives, with 707 false positives and 644 false negatives. This operating point increases recall compared to the GRU-only threshold of 0.88 while maintaining comparable precision, which is helpful when the goal is to capture more at-risk customers.

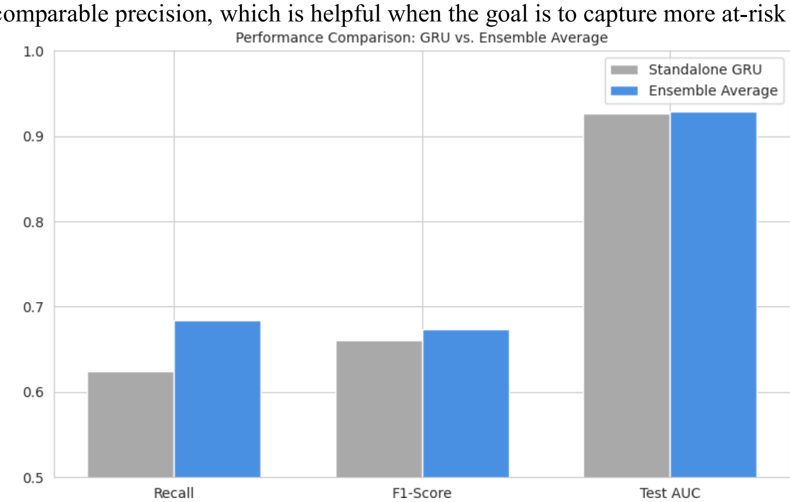

Figure 6: Comparative bar chart of GRU vs. Ensemble Average performance metrics.

To demonstrate the effectiveness of the proposed approach, Figure 6 provides a comparative analysis between the standalone GRU model and the ensemble average. The results indicate that the ensemble method achieves improved stability and better identification of churn customers compared with the individual model. During evaluation, the TensorFlow runtime reported inference step times between 6 and 24 ms, indicating that the recurrent ensemble model is computationally efficient for real-time applications. This end-to-end implementation provides a practical pipeline from data preparation to deployable churn prediction.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

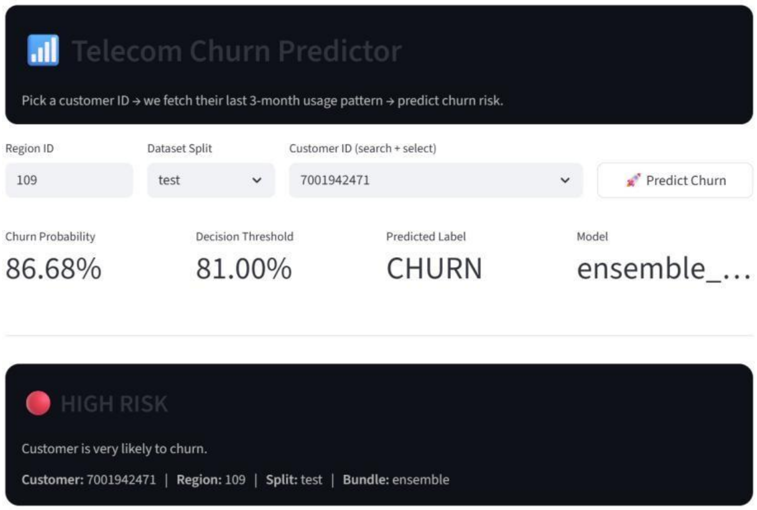

Figure 7: User interface of the Streamlit-based telecom churn prediction dashboard.

The practical implementation of the proposed system is illustrated in Figure 7, which presents the interactive Streamlit dashboard. The interface allows users to select a Region ID and retrieve subscriber information using a Customer ID. For example, in the demonstration for Region 109, the system automatically retrieves the customer's behavioral sequence and applies the ensemble model to compute the churn probability. In this case, the predicted probability is 86.68%, which exceeds the optimized decision threshold of 0.81. Therefore, the system classifies the customer as CHURN and generates a high-risk alert. This interface demonstrates how the proposed churn prediction framework translates sequential feature modeling and optimized decision thresholds into a practical decision-support tool for telecom customer retention strategies.

V. CONCLUSION

This paper presented a complete churn prediction workflow utilizing region-specific recurrent models, probability-level ensembling, and validation-based threshold optimization. In the experimental evaluation focused on Region 109, an equal-weight ensemble of RNN, LSTM, and GRU architectures demonstrated superior discrimination capability, achieving a test AUC of 0.9293 and a decision- threshold performance of 0.6736 F1 at an optimized operating point of 0.81. The deployment architecture successfully separates model serving via a Flask REST API from user interaction through a Streamlit interface, offering both customer-ID based lookups and manual-entry experiences. This dual-layer design makes the system highly suitable for both academic demonstration and practical prototyping within the telecommunications sector. Future research directions will explore the integration of longer sequence lengths, transformer-based architectures, and the calibration of predicted probabilities. Accurate churn prediction systems enable telecom providers to implement proactive retention strategies and reduce customer attrition. Additionally, implementing cost-sensitive threshold selection strategies aligned with specific retention campaign budgets could further enhance the operational utility of the proposed pipeline.

VI. ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to their respected guide Jeevan Babu Maddala for the continuous support, valuable suggestions, and insightful guidance throughout the course of this work. His encouragemesnt and expertise greatly contributed to the successful completion of this article.
We are also thankful to the Project Coordinator, Dr. N. Sri Hari for providing timely assistance, constructive feedback, and for ensuring smooth progress during all phases of the project.

Our heartfelt thanks go to the Head of the Department, Dr. V. Ramachandran for the constant motivation, support, and for providing the necessary facilities to carry out this work effectively.

We extend our deep appreciation to the Principal, Dr. Y. Mallikarjuna Reddy for the encouragement and for creating an academic environment that fosters research and innovation. Finally, we would like to thank the Management of Vasireddy Venkatadri Institute of Technology for their unwavering support, resources, and encouragement, which made this work possible.

REFERENCES

[1] W. Verbeke, D. Martens, C. Mues, and B. Baesens, "Building comprehensible customer churn prediction models with advanced rule induction techniques," Expert Systems with Applications, vol. 38, no. 3, pp. 2354–2364, 2011. (Context: churn modeling foundations).

[2] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] K. Cho et al., "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," arXiv:1406.1078, 2014.

[4] J. Mena and F. Widiputra, "Evaluation of Recurrent Neural Networks and Transformers for Customer Churn Prediction," 2024. (Accessible preprint/working paper).

[5] M. Kumar et al., "Hybrid neural network approach for telecom churn prediction," Scientific Reports, 2024.

[6] P. Prabadevi et al., "Enhancing customer churn prediction in telecommunications using a deep learning approach," Results in Engineering, 2023.

[7] A. Bogaert and S. Delaere, "Ensemble Methods in Customer Churn Prediction: A Comparative Analysis," Mathematics, vol. 11, no. 3, 2023.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.

[9] J. Davis and M. Goadrich, "The Relationship Between Precision–Recall and ROC Curves," in Proc. ICML, 2006.

[10] T. Saito and M. Rehmsmeier, "The Precision–Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets," PLoS ONE, vol. 10, no. 3, 2015.

[11] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv:1412.6980, 2014.

[12] J. De Caigny, K. Coussement, and K. W. De Bock, "A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees," European Journal of Operational Research, 2018.

[13] S. Huang, N. Kechadi, and B. Buckley, "Customer churn prediction in telecommunications," Expert Systems with Applications, vol. 39, no. 1, pp. 1414–1425, 2012.

[14] Y. Xie, X. Li, E. W. T. Ngai, and W. Ying, "Customer churn prediction using improved balanced random forests," Expert Systems with Applications, vol. 36, no. 3, pp. 5445–5449, 2009