CHURNGUARD-RAG: SHAP-EXPLAINABLE CHURN PREDICTION WITH RETRIEVAL-AUGMENTED RETENTION ADVISORY

[1]Jyothi G G, [2]Dr. Chetana Prakash, [3]Dr. Sreenivasa B R
[1]M.Tech. Student, [2]Professor, [3]Professor and Head
[1]Department of Artificial Intelligence and Data Science,
[1]Bapuji Institute of Engineering and Technology, Davanagere, Karnataka, India

Abstract : For telecom operators, subscriber attrition has grown into one of the most pressing business concerns, since every customer who leaves takes a direct bite out of revenue and long-term growth. Spotting these at-risk subscribers ahead of time gives providers a window to intervene with the right retention offer before the account is lost. This paper introduces ChurnGuard-RAG, a framework built around three components working together: ensemble-based machine learning for prediction, SHAP for explaining those predictions, and a retrieval-augmented (RAG-inspired) module for turning explanations into concrete retention advice. The framework was tested against the IBM Telco Customer Churn dataset. Once the data had been cleaned, enriched with engineered features, filtered down to the most informative attributes, and rebalanced using SMOTE, four ensemble classifiers were trained side by side for comparison. Of these, Random Forest delivered the strongest all-round numbers — 76.87% accuracy, 72.31% recall, and an F1-score of 62.34%. SHAP was then applied to surface the specific factors driving each churn prediction, and the RAG-inspired component used those factors to pull the closest-matching retention policy from a knowledge base, producing a tailored recommendation for each customer. Taken together, the results indicate that this framework can help telecom providers flag high-risk subscribers and match them with a suitable retention response.
***IndexTerms* - churn prediction, ensemble learning, RAG-inspired decision support, retention recommendation, SHAP explainability**

I. INTRODUCTION

Holding on to an existing subscriber costs telecom operators far less than winning a new one, which is why churn continues to sit near the top of industry concerns [1], [2]. When a provider can flag likely leavers early, it buys time to act and protects revenue that would otherwise be lost. Machine learning has pushed prediction accuracy forward considerably, yet most published models stop at a yes/no churn flag and offer nothing on what to do next for that customer. In practice, this leaves frontline staff falling back on gut instinct or generic playbooks when deciding how to respond. ChurnGuard-RAG was built to close that gap: it brings together ensemble-based prediction, SHAP explainability, and a Retrieval-Augmented Generation (RAG)-inspired module for decision support. Four ensemble classifiers are trained and benchmarked against one another on SMOTE-balanced data, with domain-specific engineered features added to sharpen predictive power. SHAP is then layered on top to pinpoint exactly which factors are pushing an individual customer toward churn [6]. Those factors feed into the RAG-inspired module, which searches a policy knowledge base with TF-IDF and cosine similarity to find the closest match, then assembles a recommendation covering the retention offer, a script the agent can use, and how urgently the customer needs to be contacted [7]. Pairing reliable prediction with an explanation and a concrete next step is what lets this framework connect churn analytics to actual retention practice.

II. LITERATURE REVIEW

A substantial body of work has applied machine learning to telecom churn, and ensemble approaches in particular have repeatedly delivered solid results. Sikri et al. [1] paired ensemble learning with SHAP in 2024 and reported 79.2% accuracy alongside 51.3% recall, showing that adding transparency to a model does not have to come at the cost of performance. Around the same time, Wagh et al. [2] benchmarked a range of classifiers and reached 80.1% accuracy through an adaptive ensemble-voting scheme. Both studies pushed prediction quality higher, but neither ventured beyond identifying at-risk customers into recommending what should actually be done to keep them.

Class imbalance is a persistent complication in this space too, since churners are typically a small minority against the base of retained customers. Chawla et al. [3] addressed this back in 2002 with SMOTE, and it is still one of the go-to preprocessing techniques for rebalancing churn data today. On the modeling side, Chen and Guestrin [4] gave the field XGBoost in 2016, and Dorogush et al. [5] followed with CatBoost in 2018 — both have since proven themselves on structured telecom datasets. Lundberg and Lee [6] contributed SHAP in 2017, which remains a dependable way to explain what a model is doing at the level of an individual prediction. A few years later, Lewis et al. [7] put forward Retrieval-Augmented Generation in 2020, and Gao et al. [8] mapped the growing RAG landscape in a 2024 survey. What stands out across this body of work is that hardly anyone has brought prediction, model transparency, and retrieval-driven recommendations into one unified system — that gap is exactly what ChurnGuard-RAG is built to close, drawing all three strands together to produce retention advice that is personalized and immediately usable.

III. PROPOSED METHODOLOGY

This section walks through how ChurnGuard-RAG is put together for the task of predicting telecom churn and producing tailored retention advice. Four building blocks make up the framework: preprocessing and feature engineering of the input data, ensemble models for the churn prediction itself, SHAP for explainability, and a RAG-inspired module that handles decision support. Figure 1 lays out how these pieces connect end to end.

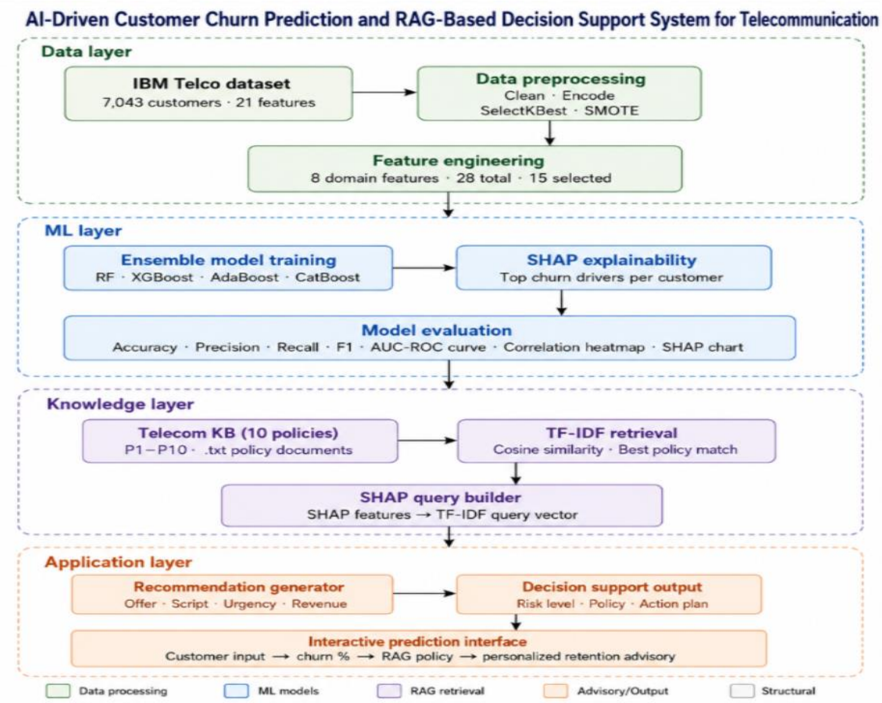

figure 1. system architecture of the ai-driven customer churn prediction and rag-based decision support system for telecommunication

3.1 Dataset

Evaluation was carried out on the IBM Telco Customer Churn dataset [9], which holds 7,043 customer records spread across 21 attributes covering demographics, account details, and the telecom services each customer subscribes to. The label to be predicted marks whether a given customer churned or stayed with the provider. Churners make up roughly 26.4% of the dataset, so the class split is moderately, though not severely, imbalanced.

3.2 Data Preprocessing

Before any model training took place, the dataset went through a cleanup pass to tighten up data quality. Eleven records were missing a TotalCharges value — all of them customers with zero tenure who had not yet been billed — and these gaps were filled using the median, which holds up better than the mean when outliers are present. Twenty-two duplicate rows were also found and dropped, bringing the working dataset down to 7,021 distinct customers. The SeniorCitizen field, originally stored as a binary flag, was recast as a categorical label to line up with the rest of the categorical columns, and every categorical feature was then label-encoded into numeric form ahead of training.

3.3 Feature Engineering

Eight additional features, grounded in telecom domain knowledge, were engineered from the raw attributes to give the models more to work with. Each one captures an aspect of customer behavior or account economics that is not directly present in the original columns:

- ChargePerTenure: computed as TotalCharges divided by (tenure + 1), reflecting how much a customer has spent relative to how long they have been subscribed.
- HighValueCustomer: flags customers whose MonthlyCharges run above 70 dollars.
- IsNewCustomer: marks customers with under six months of tenure.
- IsLongTermCustomer: marks customers with more than 24 months of tenure.
- ContractRisk: a numeric risk score assigned by contract type — 3 for month-to-month, 2 for one-year, and 1 for two-year agreements.
- HasPremiumServices: flags customers subscribe to at least one of OnlineSecurity, TechSupport, or DeviceProtection.
- ServiceCount: a simple tally of how many telecom services the customer subscribes to.
- ChargePerService: the average monthly spend per subscribed service.

3.4 Feature Selection

Once encoding was complete, the feature set stood at 27 input variables. SelectKBest paired with an ANOVA F-test was used to narrow this down to the 15 most informative features for predicting churn, trimming away attributes that contributed little while keeping those most tied to the target. Notably, five of the eight engineered features survived this cut, which is a reasonable indicator that the feature engineering step was adding real value rather than noise.

3.5 Class Imbalance Handling

With churners accounting for only 26.4% of the data, the training set carried a moderate imbalance. SMOTE was applied strictly to the training portion, to avoid leaking synthetic patterns into evaluation; the technique synthesized enough additional churn examples to take the balanced training pool up from an original 5,616 records to 8,262. Recall was treated as a priority metric here, since missing a customer who was actually about to churn means losing them for good with no second chance.

3.6 Classifier Training

Four ensemble algorithms went head to head in this study: Random Forest, XGBoost, AdaBoost, and CatBoost. Random Forest pools the votes of many decision trees, which steadies predictions and keeps overfitting in check. XGBoost takes a different route, growing trees one after another, so each new tree corrects mistakes left by the last. AdaBoost leans into previously misclassified examples, which tends to make it more alert to churn cases specifically. CatBoost, meanwhile, handles categorical fields natively and needs comparatively little preprocessing to perform well. Every model was scored on the same stratified 80:20 train-test split for a fair comparison.

3.7 SHAP Explainability

SHAP was brought in to make the models' reasoning legible, both at the level of overall model behavior and for single customer predictions. It works by attaching a contribution score to each feature, which makes it possible to point to exactly which factors are driving a given churn outcome. These per-feature contributions do not just aid interpretation — they are also passed downstream to the decision support module.

3.8 RAG-Inspired Decision Support

The last piece of the pipeline turns predictions into personalized retention recommendations through a RAG-inspired approach. Ten retention policies are kept as structured text documents in a knowledge base, and each one specifies its trigger conditions, supporting rationale, a recommended offer, a script for the agent, an urgency rating, and the outcome that can be expected. At prediction time, the SHAP features carrying the most weight are turned into a TF-IDF query, which is matched against the knowledge base using cosine similarity to surface the best-fitting policy. That policy is then packaged into a structured recommendation that gives customer service staff a concrete, ready-to-use retention action.

IV. RESULTS AND DISCUSSION

4.1 Model Performance Comparison

Table 1 lays out how each of the four classifiers performed on the held-out test set, with the top score in every column shown in bold. Taking F1 and recall together, Random Forest comes out as the most well-rounded of the four, posting an F1 of 62.34 percent on top of 72.31 percent recall. Look at recall on its own, though, and AdaBoost takes the lead at 81.45 percent — arguably the number that matters most, since it reflects how many likely churners actually get flagged. CatBoost, for its part, wins on the other two headline numbers, reaching 76.80 percent accuracy and an AUC of 0.8356. XGBoost sits in a respectable middle ground on accuracy and precision but falls behind everyone else on recall.

table 1. classification results: four smote-balanced ensemble models on ibm telco test set

Model	Accuracy	Precision	Recall	F1-Score	AUC
Random Forest + SMOTE	76.87%	54.79%	72.31%	62.34%	0.8340
XGBoost + SMOTE	77.37%	56.05%	67.20%	61.12%	0.8322
AdaBoost + SMOTE	73.81%	50.33%	81.45%	62.22%	0.8335
CatBoost + SMOTE	76.80%	54.91%	69.09%	61.19%	0.8356

4.2 ROC-AUC Comparison

Figure 2 overlays the ROC curves of all four models, and every one of them clears the diagonal reference line for random guessing (AUC 0.50) with room to spare. What is striking is how closely bunched the curves are, all landing somewhere between 0.83 and 0.84 AUC — no one classifier is pulling meaningfully ahead of the others at telling churners apart from loyal customers. Given how different these four algorithms are under the hood, that convergence points to the feature engineering pipeline doing most of the heavy lifting, and suggests that squeezing out further gains would likely depend on richer data or new features rather than swapping in a different model family.

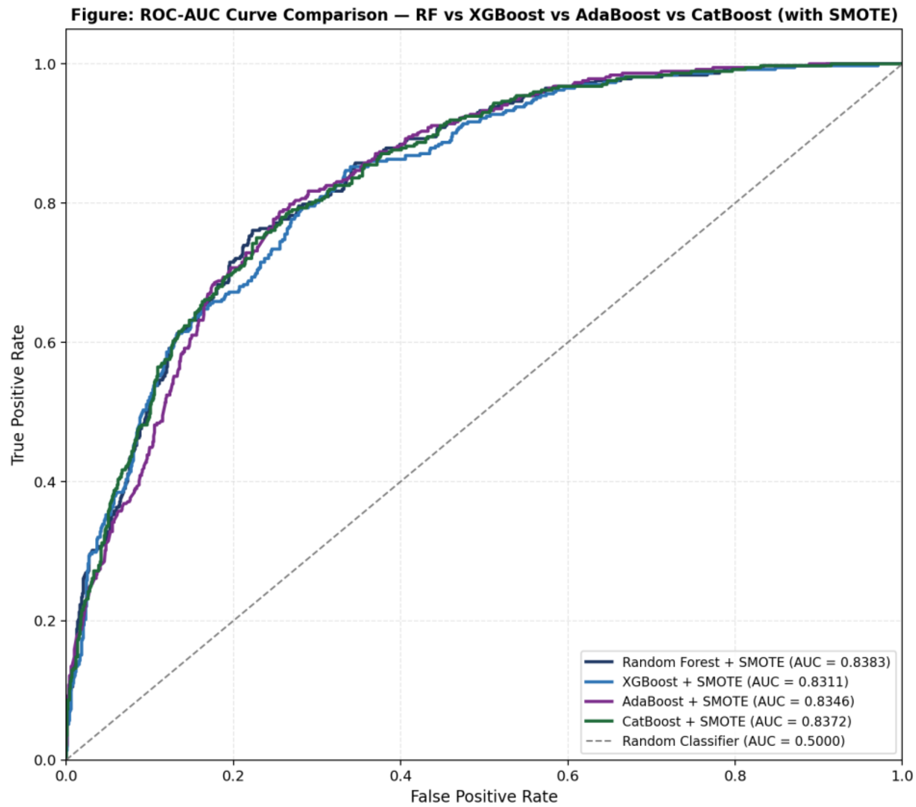

figure 2. roc-auc curves for all four classifiers with smote balancing

4.3 Correlation Analysis

Figure 3 shows the correlation heatmap spanning the 15 selected features and the churn label. ContractRisk carries the strongest positive correlation with churn at 0.40 — in plain terms, month-to-month subscribers are considerably more likely to leave than those locked into longer agreements. IsNewCustomer is not far behind at 0.31, which lines up with the idea that a customer's earliest months are when they are most likely to walk away. On the flip side, tenure shows the strongest negative correlation at -0.35, consistent with the notion that loyalty tends to build the longer someone stays. The engineered ChargePerService feature also shows up with a correlation worth noting, lending support to its place in the model. One number that looks alarming but is not is the near-perfect negative relationship between Contract and ContractRisk — that is simply because ContractRisk was built directly from the Contract column in the first place, so the two were always going to move in lockstep.

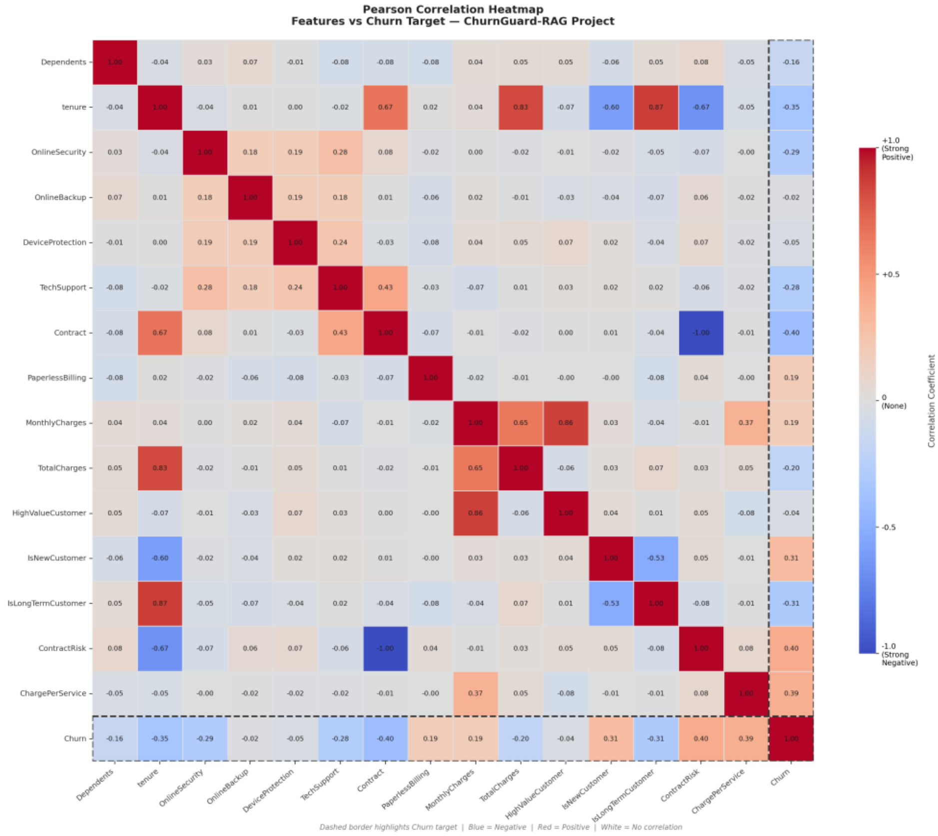

figure 3. pearson correlation heatmap of 15 selected features and churn target

4.4 SHAP Feature Importance

Figure 4 ranks the ten features that carry the most weight in the Random Forest model, scored by mean absolute SHAP value. Contract type sits well clear of everything else as the single biggest driver, at 0.8180 — more than double the next-ranked feature. Tenure comes in second at 0.4581, with MonthlyCharges close behind at 0.4258. Fourth place goes to the engineered ChargePerService feature at 0.2037, which is worth flagging since it did not exist in the raw data at all — its ranking here is good evidence that the domain-informed feature construction contributed real predictive signal rather than just padding the feature count. OnlineSecurity closes out the top five at 0.1896, matching the broader pattern that customers without protective add-on services tend to be more churn-prone.

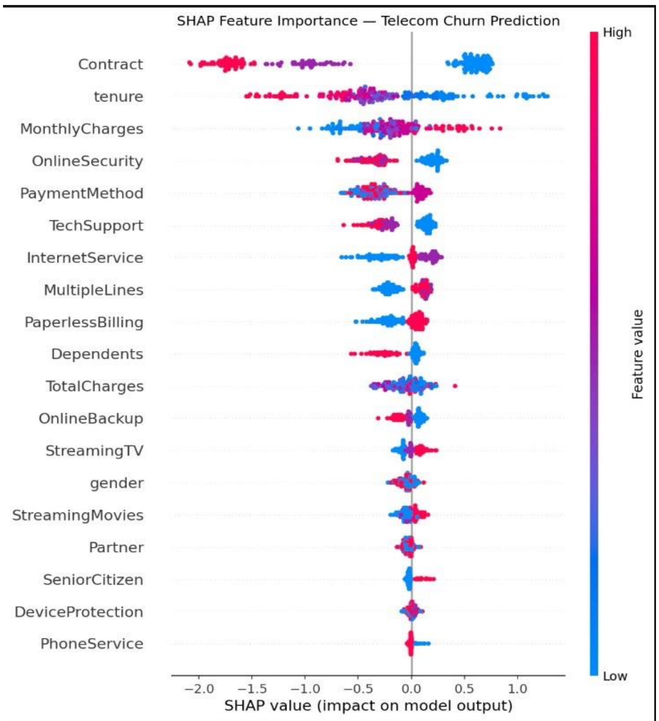

figure 4. shap feature importance: top 10 churn drivers from random forest model

4.5 RAG Knowledge Base

Table 2 lists the ten retention policies held in the ChurnGuard-RAG knowledge base. Each one was drawn up based on patterns found in the IBM Telco data together with general telecom industry knowledge about what drives churn and what actually works for retention. The set spans a wide range of retention angles — from managing contract-related risk and pricing complaints to welcoming new sign-ups, getting customers enrolled in tech support, resolving fiber-quality issues, looking after senior customers, packaging streaming bundles, keeping billing transparent, rewarding customers who have stayed a long time, and pulling several services together into one deal.

table 2. churnguard-rag knowledge base — ten telecom retention policies

ID	Policy Name	Recommended Offer
P1	Contract Retention	20% discount for upgrading to 12-month annual contract
P2	High Charges Optimization	15% loyalty discount plus free service usage audit
P3	Early Churn Prevention	Dedicated onboarding specialist assigned for 90 days
P4	Tech Support Enrollment	3 months free premium tech support trial
P5	Fiber Quality Retention	Free network audit and router upgrade at no charge
P6	Senior Customer Care	Dedicated senior support line with 25% loyalty discount
P7	Streaming Bundle Offer	Bundle discount of 25% combining internet, phone and streaming
P8	Billing Transparency	Free bill review and one-time credit of 25 dollars
P9	Loyal Customer Reward	30% discount for long-term customers with premium status
P10	Multi-Service Consolidation	Bundle all active services with 20% total discount

4.6 Sample System Output

Table 3 walks through the system's end-to-end output for five sample customers, with the advisory column showing the full personalized recommendation the RAG module produced once it had matched each profile to its closest policy.

table 3. customer risk classification and retention advisory

Customer	Charges	Tenure	Churn %	Risk	Advisory Generated by RAG Module
CUST_001	$95.50	2 mo.	82%	Critical	Offer 20% discount for annual upgrade. Contact within 24 hours.
CUST_002	$45.00	24 mo.	55%	High	Enroll in 3 months free tech support. Contact within 3 days.
CUST_003	$110.00	4 mo.	91%	Critical	Offer 15% discount and plan audit. Contact immediately.
CUST_004	$35.00	60 mo.	2%	Low	Proactive loyalty reward. Standard scheduled follow-up.
CUST_005	$88.00	1 mo.	76%	Critical	Assign onboarding specialist and offer contract upgrade.

4.7 Real-Time Churn Prediction and Retention Advisory

A terminal-based application was built to put the framework through its paces in something closer to a real-time setting. It takes in customer details, scores the churn risk, explains that score through SHAP, pulls the best-matching retention policy from the knowledge base, and hands back a personalized recommendation, all in one pass. Figure 5 captures a sample run. Alongside the churn probability, the output spells out why the model reached that conclusion and lays out a suggested offer, an agent script, an urgency rating, and an estimate of the revenue at stake. Taken together, this demonstrates that the framework is capable of supporting telecom operators with retention decisions that are both timely and well-informed.

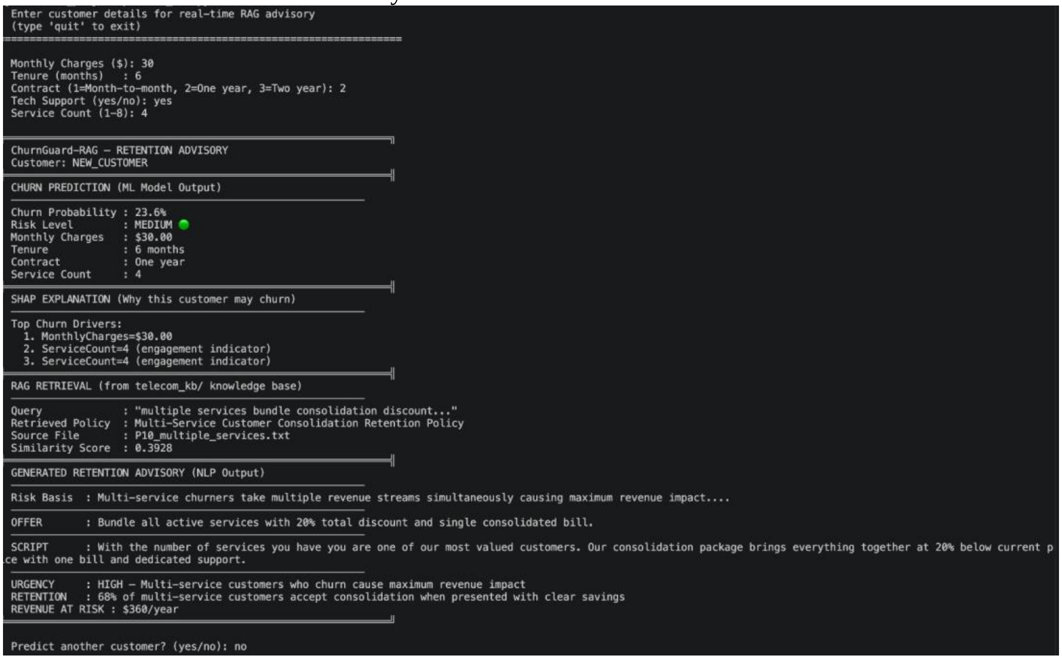

figure 5. real-time output of the proposed churnguard-rag framework

V. CONCLUSION

This paper walked through ChurnGuard-RAG, an approach that ties together three things for telecom retention: prediction from an ensemble model, an explanation of that prediction via SHAP, and a retrieval-driven advisory step that turns the explanation into something an agent can act on. Adding eight domain-informed features to the IBM Telco benchmark lifted performance across every one of the four classifiers tested. Of the four, Random Forest paired with SMOTE gave the steadiest all-round numbers — accuracy of 76.87 percent, recall of 72.31 percent, and an F1-score of 62.34 percent — whereas AdaBoost pulled ahead specifically on recall (81.45 percent) and CatBoost specifically on AUC (0.8356). Across the SHAP analysis, contract type, tenure, and monthly charges kept surfacing as the dominant churn predictors, and ChargePerService — an engineered feature with no counterpart in the raw data — landing in fourth place is a fair sign that the feature-construction work paid off rather than just adding noise columns. The part of this system that goes beyond most existing work is the advisory module: it does not stop at a probability score but hands back a retention recommendation, grounded in an actual policy, in real time. That is the piece prediction-only systems leave out entirely. From here, the natural next steps are moving retrieval over to a dense vector approach such as FAISS or ChromaDB, letting a large language model generate the advisory text instead of filling a fixed template, and running the same pipeline against other telecom datasets to see how well it holds up outside this one benchmark.

VI. FUTURE SCOPE

- Swap the current cosine-similarity retrieval for FAISS or ChromaDB-based dense vector search to scale to larger policy collections.
- Bring in a lightweight, quantized LLM (LLaMA-3 served through the Groq API is one candidate) to write the advisory text freely rather than filling in a fixed template.
- Run the same pipeline on other public telecom churn datasets — Orange Telecom and Cell2Cell are two obvious candidates — to see whether the results hold up beyond IBM Telco.
- Develop a Streamlit or Flask web dashboard so the system can plug into CRM workflows for real-time use.
- Explore federated learning so multiple operators can train jointly without exposing raw customer data to one another.
- Incorporate temporal modeling, using LSTM or Transformer architectures, to capture patterns in sequential usage history.

ACKNOWLEDGMENT

The authors thank the Department of Artificial Intelligence and Data Science, Bapuji Institute of Engineering and Technology, Davanagere, for the infrastructure and support provided during this work.

REFERENCES

[1] A. Sikri, R. Jameel, S. M. Idrees, and H. Kaur, "Enhancing customer retention in telecom industry with machine learning driven churn prediction," Scientific Reports, vol. 14, p. 13011, 2024.

[2] S. K. Wagh, A. A. Shinde, P. R. Patil, and D. S. Kolte, "Customer churn prediction in telecom sector using machine learning techniques," Results in Control and Optimization, vol. 14, p. 100342, 2024.

[3] 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.

[4] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794, 2016.

[5] A. Dorogush, V. Ershov, and A. Gulin, "CatBoost: gradient boosting with categorical features support," arXiv preprint arXiv:1810.11363, 2018.

[6] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," Advances in Neural Information Processing Systems, vol. 30, 2017.

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," Advances in Neural Information Processing Systems, vol. 33, pp. 9459-9474, 2020.

[8] Y. Gao, Y. Xiong, X. Gao, K. Jia et al., "Retrieval-augmented generation for large language models: a survey," arXiv preprint arXiv:2312.10997, 2024.

[9] IBM Sample Data Sets, "Telco Customer Churn Dataset," IBM Developer, 2019. Available: https://developer.ibm.com/patterns/predict-customer-churn-using-watson-studio-and-jupyter-notebooks/. Accessed June 2026.

[10] R. Ahmed, M. Al-Masri, and K. Samara, "Enhancing customer churn prediction in telecommunications: an adaptive ensemble learning approach," arXiv preprint arXiv:2408.16284, 2024.

[11] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[12] A. El Attar and M. El-Hajj, "Explainable AI-driven customer churn prediction: a multi-model ensemble approach with SHAP-based feature analysis," Frontiers in Artificial Intelligence, vol. 9, 2026.