Executive Summary

15.572 Analytics Lab Final Project
Team JABY
Joshua Cai, Annie Liu, Brian Lu, Yi Wang
Sponsored by Qrious Insight

Problem Statement

Qrious Insight relies on a 150,000-person panel to generate behavioral intelligence for clients, but a subset of users attempts to obtain incentives through fraudulent or automated behavior. Existing rule-based checks catch only known patterns, leaving more subtle fraud undetected. The core challenge of this project was to develop a machine learning system that identifies previously unseen fraudulent users by analyzing their behavioral signatures, while maintaining ethical standards that avoid reliance on sensitive demographic data.

Approach and Methods

Our approach focused on transforming high-resolution web and app activity into structured temporal representations. After correcting a significant sampling bias by realigning all users' activity windows to a standardized calendar-month format, we tested a range of models, including XGBoost, deep neural networks, CNNs, transformers, and fine-tuned BERT models. Multiple combinations of representations and window structures were evaluated, and the strongest results came from categorized web activity using 30-day windows and daily time resolution. Fraudulent users consistently displayed higher, more erratic browsing patterns across categories such as Search Engines, Gambling, Adult Content, and obscure "Unable to Scrape" domains, confirming that behavior provides a rich and reliable signal.

Result

The final activity-based XGBoost model achieved an AUC of approximately 71% and represents the most ethically robust and operationally appropriate solution. Although the Append data (user activity summary plus demographic information) model reached an AUC of 81%, we explicitly advise against adopting it as a primary decision system due to the ethical and reputational risks associated with demographic inference. The Append model is best used only as a reference benchmark to contextualize performance ceilings.

Insights, Impact, and Conclusion

Operationally, we recommend deploying the activity model within a recurring 30-day fraud-screening cycle, supplemented by human-verified active learning to validate high-probability fraud predictions and iteratively strengthen the model. Applying this system to the TBD population enables Qrious Insight to detect and flag a significant number of future potential fraud cases, improving data integrity and reducing operational costs. With a panel of this size, removing unrepresentative users yields an estimated annual savings of approximately $250,000 while significantly enhancing the quality and reliability of insights delivered to clients.

This project demonstrates that behavioral modeling, grounded in ethically sound data use, offers Qrious scalable, defensible, and cost-efficient path to substantially stronger fraud detection moving forward.

Behavioral Fraud Modeling: Separating Honest Patterns from Fraudulent Chaos

15.572 Analytics Lab: Action Learning Seminar on Analytics, Machine Learning, and the Digital Economy

Joshua Cai
jcai2701@mit.edu

Annie Liu
yil297@mit.edu

Brian Lu
luki90@mit.edu

Yi Wang
yw6157@mit.edu

1. Introduction

This project, sponsored by Qrious Insight, aims to identify fraudulent user behavior within a high-resolution behavioral tracking dataset containing detailed temporal and contextual information. Our workflow begins by decomposing the raw data into semantically meaningful topic groups, followed by preprocessing user activity sequences at multiple temporal granularities. We then conduct exploratory data analysis (EDA) across each data type to characterize systematic differences between fraud and non-fraud users and to guide feature engineering.

After generating several candidate representations of the dataset, we initially focused on a 14-day activity window and later examined an hourly-resolution formulation. To reduce sampling bias and improve the robustness of downstream modeling, we applied a temporal reordering procedure that preserves behavioral structure while mitigating distributional imbalance. We next trained multiple baseline models on the primary dataset and evaluated them across all alternative data formulations. Among these models, XGBoost achieved the strongest performance, reaching an AUC of approximately 71% for fraud detection.

As the primary deliverable, we provided Qrious Insight with the trained model and accompanying documentation, enabling their team to determine an operational threshold through internal back-testing. For future work, we proposed implementing a reinforcement loop that periodically retrains the model as new behavioral data become available. Additionally, we recommended establishing a recurring fraud-screening workflow that evaluates all users on a rolling 30-day basis.

2. Problem Statement

Qrious Insight collects detailed mobile phone activity data from a panel of users—including website browsing history, app usage patterns, battery activity, and other behavioral signals—in exchange for incentives. Panelists also complete client-specific surveys that Qrious monetizes by providing aggregated insights. While the majority of participants contribute valid and representative data, a subset of users attempt to “game the system” to obtain incentives without providing meaningful human-generated behavior. Such unrepresentative—or effectively fraudulent—users may employ automated bots, autoclicker applications, or create multiple accounts using different emails or phone numbers to increase their earnings.

To mitigate these issues, Qrious Insight currently applies a set of deterministic rules to flag and remove known fraudulent accounts. Examples of these non-exhaustive rules include: identifying users who register with multiple emails, users associated with multiple phone numbers, and users who visit or install from predefined lists of fraudulent websites or applications. Although these rules capture a substantial portion of problematic users, they do not fully address more subtle or emerging forms of unrepresentative behavior.

In this project, we hypothesize that there exist underlying behavioral patterns—reflected in the high-resolution activity data—that systematically differentiate clean users from unrepresentative users. The objective is therefore to improve overall data quality by detecting these previously unflagged users through machine learning. To that end, we evaluate multiple dataset formulations and modeling approaches, with the goal of identifying the most informative data representation and the most effective predictive model for capturing these latent behavioral differences.

3. Data

The entire panel with Qrious Insight consists of around 150,000 users, and the panel we worked with is a subset of approximately 17,000 from the 50,000 Android users whose mobile phone activities are continuously collected by Qrious Insight. We primarily worked with three datasets: web activity, app activity, and Append. The web activity data logs each panelist's browsing behavior, including visit start and end times, website domains, and search terms when applicable. The app activity data follows a similar structure, recording each app's usage start and end times along with the app package name, which uniquely identifies the application. The Append data provides aggregated monthly behavioral information and some demographic details of the panelists, such as income tier, ethnicity, and counts of visits to various store categories. To avoid privacy risks and prevent demographic profiling, we only considered Append data when necessary and used it exclusively to construct an optional, alternative model for our sponsor.

3.1 Preprocessing

A central step in preprocessing is the construction of fraud and clean user labels. Fraudulent users are straightforward to define: any user previously flagged and banned by Qrious Insight's internal rule-based system—after human verification—is labeled as fraud. Defining clean users is less direct. Based on guidance from our sponsor, we assume a user is reliably clean if they have remained in the panel long enough without ever being flagged. Operationally, we classify as clean the top 20% earliest registrants (based on percentile of registration timestamp) who have never been flagged.

All remaining users have unknown status and are labeled as TBD (to be determined). After building the final model, we will apply it to these TBD users to generate a list of potential fraud cases for Qrious Insight to review.

This labeling process yields approximately 800 fraud users and 3,400 clean users, introducing substantial class imbalance. To address this and prepare data for window-based modeling, we introduce a hyperparameter *n* representing the temporal window length (e.g., 14 or 30 days). Users, fraud or clean, who do not have enough activity to support the required window are removed. For each fraud user, we then identify two clean users who are active both before and after the same window. To maximize the number of matched clean users, we implemented a matching algorithm that begins with fraud users who have the fewest eligible clean counterparts. After matching and filtering, our modeling dataset contains approximately 2,000 panelists prior to the train/validation/test split.

3.2 Web Data Preprocessing

The web dataset has three inherent dimensions: user, visited website, and time. To prepare the data for sequence modeling, we restructure the dataset into a temporal matrix where rows represent users and columns represent time steps.

We explored two primary representations of website information:

- **Categorized website representation:** We assign each website domain to one of 30 manually defined website categories, based on domain text. The resulting dataset consists of 30×*t* features, where *t* is the number of unique time steps.
- **Embedded domain-name representation** Using two state-of-the-art domain encoders (URLNet and DomainBERT) we embed each website domain name into a dense representation. For each time step, a user's representation is the duration-weighted average of all embedded domains visited during that time. This produces a 2D sequence: time steps by embedding dimension.

We also evaluated two window lengths:

- **14-day window**: ending at the last recorded activity prior to the ban for fraud users.
- **30-day window**: constructed analogously.

Finally, for the categorized representation, we tested two time resolutions:

- **Hourly resolution**: aggregates visit duration per website category per hour.
- **Daily resolution**: aggregates duration per category per day.

After evaluating all combinations of representation, window length, and time resolution, the best-performing dataset was:

- **Website representation**: categorized
- **Window length**: 30 days
- **Time resolution**: daily

The majority of subsequent EDA and modeling focuses on this dataset variant.

3.3 App Data Preprocessing

The preprocessing of app activity data parallels the web data approach. We use the same window-length and time-resolution settings as defined above and rely solely on categorized representations of apps. Qrious Insight provided an initial curated list mapping app packages to categories; however, this list covered only a small portion of all apps observed. To expand coverage, we applied a keyword-based categorization procedure on the package names and manually verified the assignments. This combined approach enabled us to categorize apps responsible for over 97% of all recorded app usage.

3.4 Sampling Bias Leading to Artificially Good Results

A major methodological consideration arises from the construction of fraud windows. Because the fraud window ends exactly at the user's last activity before being banned, fraud users are guaranteed to exhibit activity near the end of their window. Clean users, whose windows are anchored differently, do not display this same pattern. A model trained directly on these windows would incorrectly learn that "activity near the end of the window" is a signal of fraud, even though this is purely an artifact of sampling rather than genuine behavioral difference.

To eliminate this bias, we align all user windows to calendar months. For each 30-day window, we locate the nearest occurrence of the first day of a month within the 30-day range, split the window at this point, and recombine the segments so that every window spans a standardized calendar interval from Day 1 to Day 30. This removes the artificial activity spike and ensures fraud and clean windows are structurally comparable. A natural consequence is that the model can only be applied to full calendar-month windows, which is a reasonable restriction given the substantial reduction in sampling bias.

3.5 Append Data

The Append dataset contains highly detailed demographic and behavioral aggregates for each panelist, including ethnicity, income tier, and monthly counts of visits to major website, app, and location categories. Because the dataset also includes columns listing specific websites, apps, and locations—information too granular and ethically sensitive for modeling—we removed these detailed fields and retained only the aggregated summaries. A fraction of the predictive signal in Append data comes from demographic attributes, which pose ethical risks if used for automated decision-making. Consequently, we treat Append-based models strictly as optional auxiliary models for the sponsor, rather than as core components of our proposed fraud detection system.

4. EDA

Our central hypothesis is that fraudulent users exhibit systematically different behavioral patterns compared to clean users, particularly in their web-browsing and app-usage activity. Behavioral signals (such as the types of websites visited, the intensity of engagement, and the temporal structure of activity) tend to be harder to fabricate consistently and therefore offer stronger separation than static features like location, device battery level, or session duration. In practice, many fraudulent users operate from a fixed device setup (e.g., always plugged in, always at 100% battery, never moving across locations), making those auxiliary features less informative. For this reason, we focus our EDA on understanding how fraud and clean users differ in what they do and how they behave over time.

4.1 Fraud Users Summary

- **fraud label distribution**: We first examine the distribution of the platform's existing fraud labels (Fig.1). The dataset is highly imbalanced: the overwhelming majority of users fall under the "Ok" category, with only a small share labeled as "Banned" or "Suspected." This imbalance is typical in fraud detection and reinforces the need for careful sampling and evaluation. Among the rule-based fraud signals (Fig.2), fraud_multiple_emails is the most common, followed by fraud_phone_number and fraud_apps, suggesting that identity-level manipulation and repeated account creation are among the earliest detectable behaviors.

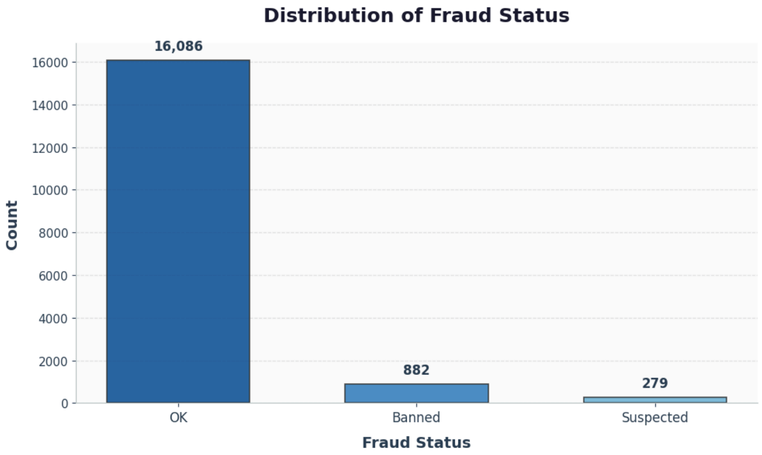

Figure 1. Distribution of Fraud Status

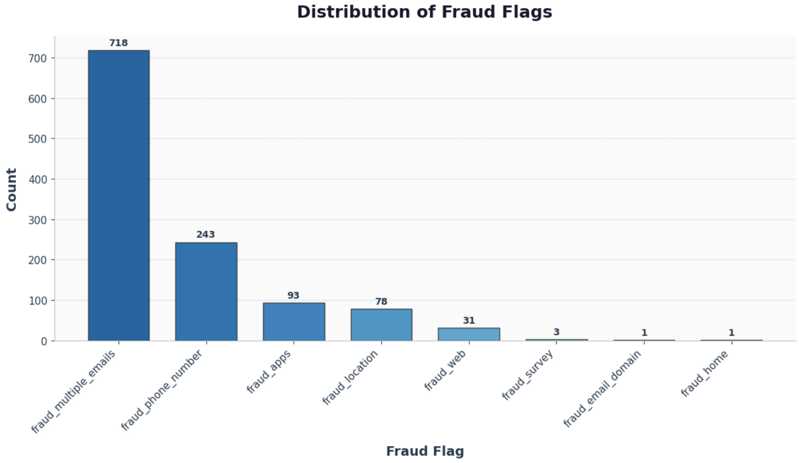

Figure 2. Distribution of Rule-Based Fraud Status

- **Time to Ban by Fraud Type**: Different types of fraud surface at different speeds (Fig.3). Users flagged for web-related violations or multiple-email fraud tend to remain in the panel for a long time, often 280–360 days, before being banned. These behaviors are subtle and harder to detect in real time. By contrast, simpler patterns such as phone-number inconsistencies and survey fraud are caught more quickly. This heterogeneity indicates that not all fraud behaviors present equally strong or immediate signals, highlighting the need for behavioral modeling.

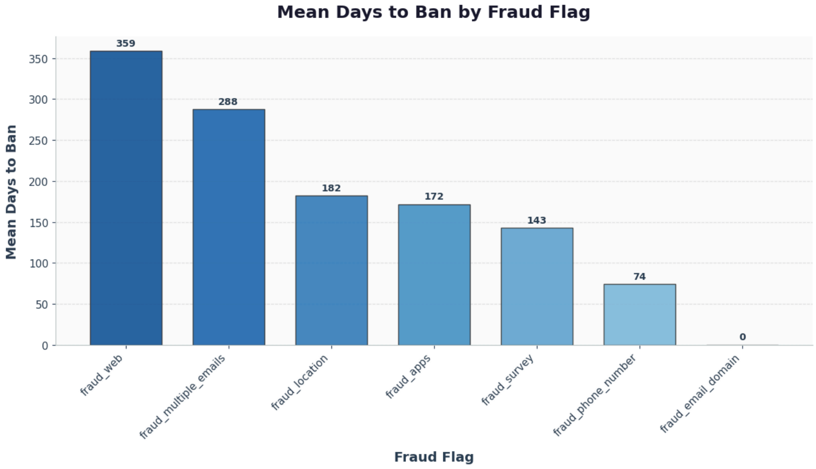

Figure 3. Mean Days to Ban by Fraud Flag

4.2 Web Browsing Behavior

To evaluate behavioral differences directly, we compare fraud and clean users' total engagement across web categories over a 30-day matched window.

- **Category-Level Differences:** Fraudulent users consistently spend more time across nearly every web category (Fig.4). The largest gaps appear in: Unable to Scrape, Search Engine, Gambling, not_top_100k, and Adult/Pornography. All of these categories show statistically significant differences. Unable to Scrape in particular shows the strongest separation, indicating that fraud users frequently interact with obscure or shielded domains. Fraud users also show elevated activity in more conventional categories such as Online Shopping, Restaurants, Technology, Finance, and Real Estate, though at a smaller scale. Clean users slightly exceed fraud users only in Movies and Streaming, and these differences are not statistically significant.

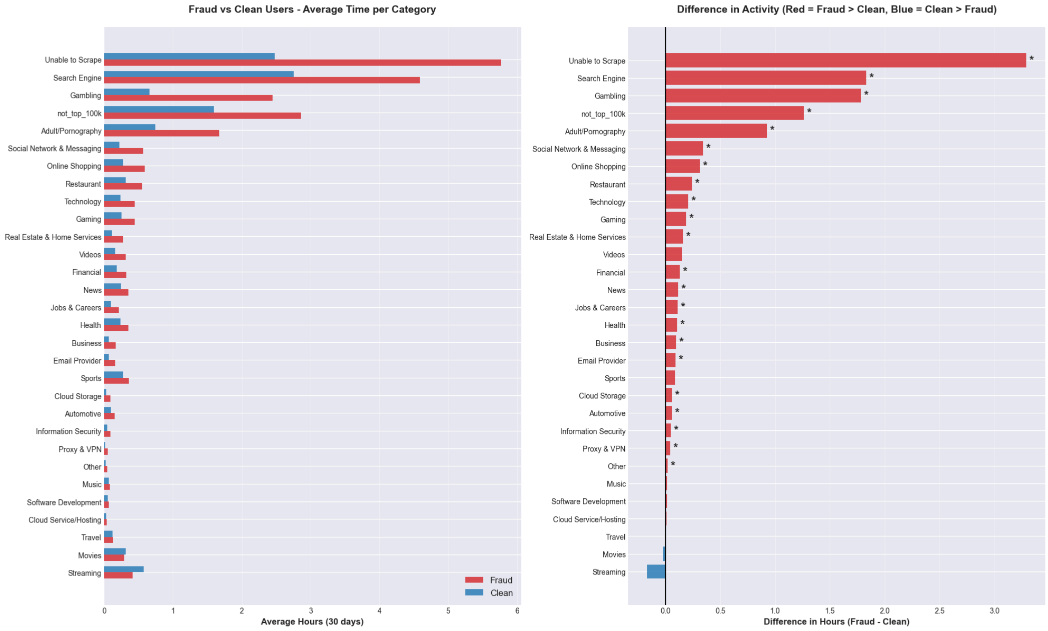

Figure 4. Web Activity Differences by Category (* = Significant Difference)

- **Temporal Patterns**: The 30-day activity curves reveal clear structural differences (fig.5). Fraud users display higher and more volatile spikes in hourly activity across top fraud-associated categories such as Unable to Scrape, Search Engine, and Gambling. They maintain elevated baselines that persist throughout the month, even outside peak periods. Additionally, they show consistent activity patterns across multiple categories, suggesting coordinated or automated browsing behaviors rather than organic usage. In contrast, clean users have stable patterns with much smaller fluctuations and lower overall engagement.

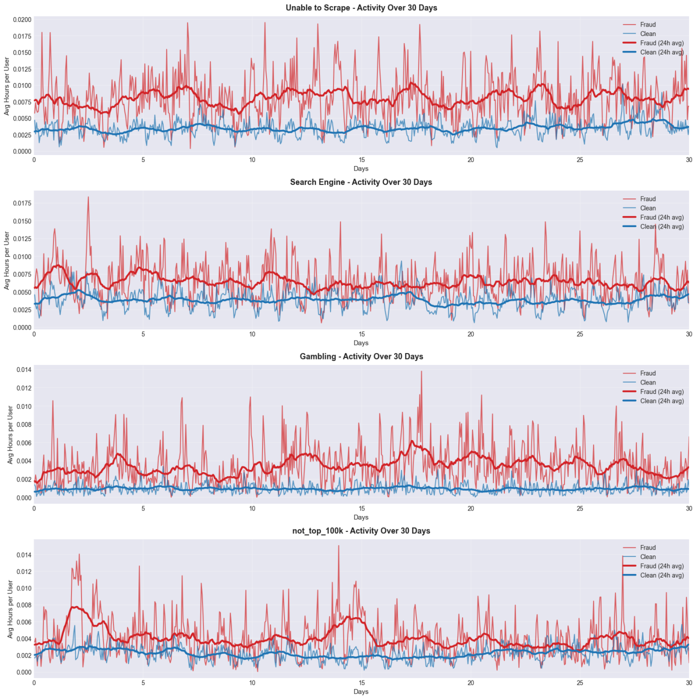

Figure 5. Fraud vs. Clean: 30-Day Activity Patterns for Top Fraud-Associated Categories

- **Outlier analysis:** Boxplots (fig.6) reveal that fraud users not only engage more on average but also exhibit substantially greater variance, with long right-tails in nearly every category. Some fraud users accumulate tens or even hundreds of hours in certain categories within a month. Clean users also display occasional outliers, but at a much smaller magnitude. While outliers contribute to the mean differences, the separation in medians confirms that the behavioral gap is not driven by a handful of extreme cases.

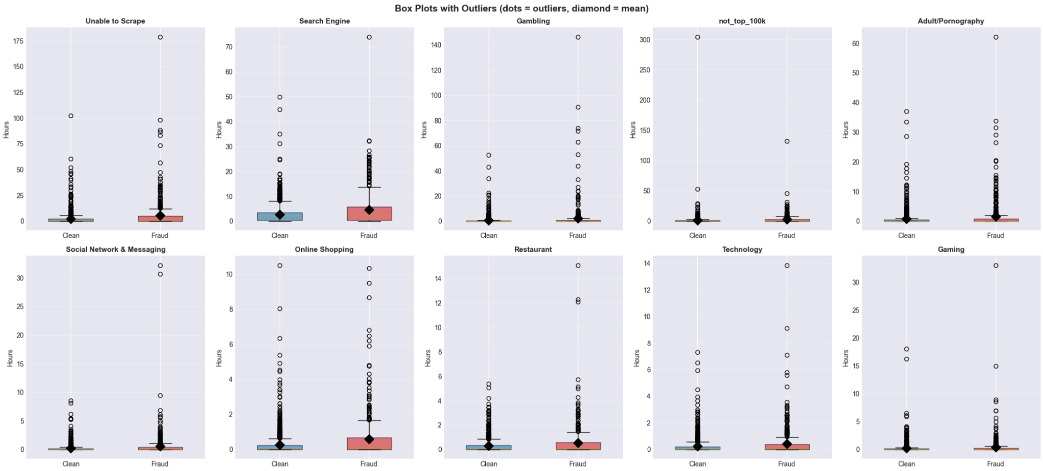

Figure 6. Category-Level Boxplots with Outliers (Clean vs. Fraud Users)

We perform the same matched analysis for app usage. However, only 4 of 20 p categories show statistically significant differences between fraud and clean users, and including app activity does not improve downstream model performance. App behavior is therefore a far weaker differentiator. For this reason, the remainder of our EDA focuses on web activity, which provides richer, more discriminative behavioral signals for fraud detection.

5. Methods

For each data modality, we evaluated a set of models tailored to the structure of the features. For the categorized web and app activity datasets, we tested both traditional and deep learning approaches, including:

- XGBoost, with systematic hyperparameter tuning
- a six-layer feedforward neural network (FNN)
- additional deep learning architectures—vanilla CNN, TabNet, and 1D CNN—which ultimately did not perform as well as the previous two models

For the domain-embedding representation of web activity, we focused on architectures suited to sequential or structured embeddings:

- a CNN, designed to capture local spatial patterns in the embedded domain sequences
- a transformer model, used to leverage longer-range dependencies across time

For the Append dataset, we explored ensemble modeling on tabular data:

- Logistic regression, random forest, and XGBoost models were used as the base for an ensemble model.
- The data includes parameters like ethnicity, age, and zip code, providing sharper insights but are sensitive variables, in addition to the activity summary variable.

For the search-term data, we applied modern NLP methods:

- fine-tuned BERT and DistilBERT classifiers, with selected layers unfrozen to better learn domain-specific semantic patterns

Although these models captured meaningful textual structure, they demonstrated limited predictive value relative to the behavioral activity models

6. Result

We evaluated multiple model architectures across all dataset formulations and found that the best-performing approach uses 30-day windows, daily time resolution, and categorized web activity features. Among all candidates, XGBoost consistently delivered the strongest performance, achieving an AUC of approximately 0.70 on the held-out test set.

6.1 Model Performance

The baseline XGBoost model achieved an AUC-ROC of **0.703**, with a balanced trade-off between identifying fraudulent users and avoiding excessive false positives. After tuning hyperparameters (including learning rate, tree depth, and column subsampling), the optimized model reached an AUC-ROC of **0.707**, a modest but consistent improvement. The ROC curves for both models lie well above the random classifier line, confirming meaningful predictive signals in the engineered behavioral features.

Confusion matrices reveal similar error profiles across the two models (fig.7). The tuned model shows a slight reduction in false negatives (44 to 49) and an increase in correctly identified fraud cases (47 to 52), reflecting a small gain in sensitivity while preserving performance on clean users. Given the scarcity of fraud cases in the matched dataset, even incremental improvements in recall translate to meaningful business value.

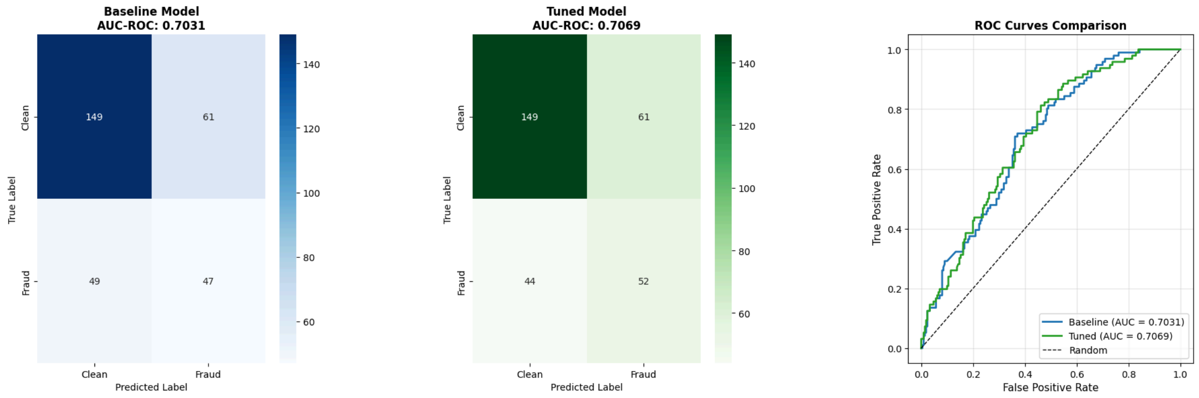

Figure 7. Model Evaluation: Confusion Matrices and ROC Curve Comparison

6.2 Feature Importance

Aggregating feature importance across daily features (fig.8) shows that no single behavior dominates the model's predictions. Instead, fraud detection relies on a combination of signals across many categories. The most influential groups include Restaurant, Technology, Online Shopping, Adult/Pornography, Search Engine, Gaming, and Gambling. These results align with our EDA: the categories where fraud users spend more time or show unusual temporal patterns tend to contribute more strongly to the model. Notably, the model does not depend only on the most extreme categories (such as Unable to Scrape). Rather, it draws from a broad set of behavioral cues, reinforcing the idea that fraud behavior appears as general deviations in activity level and structure.

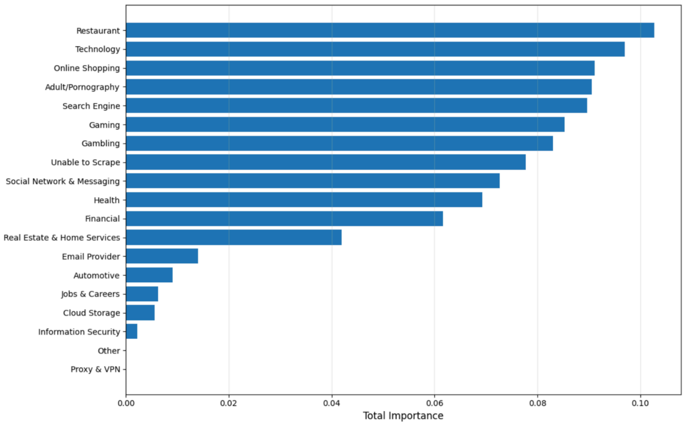

Figure 8. Feature Importance by Web Category (Summed Across Daily Features)

7. Conclusion

7.1 Business Impact and Model Deployment Strategy

As part of this project, we delivered three key assets to Qrious Insight:

1. A technically validated 30-category website taxonomy used to convert raw browsing behavior into structured activity features.
2. Two fully trained machine learning models, an activity-based model and an Append-based model, accompanied by documentation, threshold-selection guidance, and implementation notes.
3. A recommended operational workflow for ongoing model reinforcement, including a 30-day routine fraud-screening cycle and an active learning loop with human verification.

Across all experiments, the activity-based model, trained on categorized web and app behaviors, achieved an AUC of approximately 71% in detecting fraudulent panelists. The Append-based model, which includes demographic attributes, achieved a higher AUC of 81%. However, because demographic-considered predictions raise ethical concerns and risk unintended profiling, we strongly recommend not using the Append-based model as the primary decision system. Instead, it should serve only as a reference model to contextualize performance boundaries and illustrate the importance of prioritizing ethically grounded modeling choices. The activity-based model, though slightly lower in predictive performance, offers a more objective, behavior-driven, and ethically defensible approach for production use.

Looking ahead, Qrious Insight can apply the activity model to the TBD user group and manually validate high-confidence predictions to determine an appropriate operating threshold. Based on expected prevalence rates and model performance, we anticipate the system will detect and flag roughly 8% of all potential fraud cases moving forward, significantly enhancing the integrity of the panel. Integrating a monthly routine check, where every user is re-evaluated using a standardized 30-day window, ensures continuous monitoring and reduces the persistence of undetected fraud.

From a business perspective, improving fraud detection yields substantial cost savings. With over 17,000 active panelists, removing unrepresentative users reduces incentive payouts and improves data quality without the need to expand panel size. Conservatively, this translates to an estimated annual savings of approximately $250,000, while simultaneously increasing the reliability of the insights Qrious provides to clients.

7.2 Future Work

Several avenues for future enhancement remain. First, the current matching process between fraud and clean users relies solely on temporal window alignment. A more refined approach would incorporate vector-distance matching based on each user’s Append activity vectors, allowing the model to compare fraud users against clean users with more similar underlying behavior and reducing residual sampling bias.

Second, expanding the observation window beyond the current 30-day framework may offer richer behavioral signals. Longer windows, such as 60-day or 90-day sequences, combined with hourly or even minute-level granularity, could improve the temporal resolution of the model and better capture subtle behavioral inconsistencies associated with fraudulent activity.

Third, model improvement is currently limited by the scarcity of labeled fraud cases. Although this limitation is structurally tied to the nature of the business (a low number of fraudulent users is a desirable operational outcome), it presents challenges for supervised learning. To mitigate this, we recommend deploying an active learning pipeline in which the model flags uncertain or high-probability fraud cases for human verification. Newly confirmed fraud cases can then be fed back into the training set, enabling iterative model refinement while avoiding negative feedback loops.

Together, these steps can substantially strengthen Qrious Insight’s fraud detection capability, maintain ethical standards in model construction, and ensure that the system evolves effectively alongside real-world behavioral patterns.