Automated Audit of ECD Approvals Using ML and Anomaly Detection

Mihir A. Bhansali
M.S. in Applied Data Science
University of Florida
Gainesville, FL, USA
mihir.bhansali@ufl.edu

***Abstract*—This report summarises my Summer 2025 internship at Amazon, where I worked on building an automated system to review and flag unusual ECD (Exception Control Document) approvals. The goal was to identify approvals that looked similar to past denials, using a mix of classification and anomaly detection models. I trained machine learning models such as XGBoost, Isolation Forest, and a Variational Autoencoder (VAE), and applied SHAP for interpretability.**

This experience helped me apply concepts from my academic work to solve real business problems, and strengthened my interest in developing reliable, interpretable AI systems for high-impact domains.

This experience deepened my understanding of how applied machine learning systems are designed and deployed in real-world, high-stakes environments where models don't just make predictions but directly support business-critical decisions, compliance workflows, and operational risk reduction.

***Index Terms*—Amazon, anomaly detection, machine learning, interpretability, SHAP, VAE, Isolation Forest, classification**

I. Personal Reflection and Career Goals

This internship at Amazon was my first industry experience and turned out to be one of the most rewarding learning opportunities in my journey so far. I have always been curious how data science can be used to solve real-world problems, especially those that influence actual business decisions, and this project was the perfect opportunity to explore that in depth.

Working on automating the audit of Estimated Completion Date (ECD) approvals gave me a hands-on view of how machine learning systems are applied in high-stakes environments. I was exposed to challenges that go far beyond the classroom, where accuracy and validation scores are often the only focus. Here, the outputs of my models had the potential to impact compliance, internal risk management, and even stakeholder accountability.

What made the experience even more meaningful was the guidance and support I received from my mentor, manager, and colleagues. Their constant feedback pushed me to think beyond just building a 'good model' and instead focus on designing systems that are explainable, trustworthy, and usable. This shift in mindset, towards impact, transparency, and business relevance, has strongly shaped my long-term goal of working on AI systems that operate safely and reliably in the real world.

II. Project Overview and Milestones

Quarter 1: Environment Setup and Exploratory Analysis

The first phase of the internship focused on setting up the technical environment and building a strong foundational understanding of the data and systems involved. I gained access to internal tools and platforms including SIPP, Shepherd, Quip, and Merlon, and set up necessary permissions for SageMaker, S3 buckets, and pre-production ETL pipelines. I also participated in sync meetings with my manager, mentor, and several colleagues to better understand the end-to-end ECD approval process, the related ETL jobs.

In parallel, I began reviewing documentation to understand the ECD approval flow, data structure, and the relationship between issue-level and permit-level information. This helped me understand how raw ETL data is transformed and used for decision-making downstream.

During this quarter, I also kicked off exploratory data analysis (EDA). I profiled the dataset by examining feature types, missing values, and distribution ranges. I explored ECD type categories such as "No", "Yes-but", and "Yes" and segmented the data across severity levels (Critical, High, Medium, Low). Using various EDA plots and summary statistics, I tried to derive early insights—such as potential patterns across denial vs approval cases—that would inform later model development.

Quarter 2: Classification Model Development

The main goal during this phase was to develop a supervised machine learning model that could serve two purposes: (1) help us understand which features most strongly influence whether an ECD request is approved or denied, and (2) generate probability-based predictions for each request to support automated auditing and future monitoring. This would allow us to surface inconsistencies and improve the overall consistency of the ECD review process.

The first step involved creating a clean and reliable training data set. I implemented feature engineering techniques such as frequency encoding for categorical variables, standard scaling for numeric attributes, and TF-IDF transformation for textual inputs such as provided justifications. A preprocessing pipeline was built to ensure that transformations were applied consistently throughout both the training and testing phases, reducing the risk of data leakage.

To prepare the model, I performed a time-based split on the dataset to mimic future-looking predictions and ensure generalizability. An ensemble classifier built using Balanced-Bagging with XGBoost as the base model, which helped address class imbalance while maintaining high predictive performance. I used cross-validation to tune parameters and evaluate consistency.

To make the model transparent, I applied SHAP (SHapley Additive Explanations) to identify key features that influenced decisions. These insights were shared with stakeholders to guide trust and improve understanding of the logic of the model. Overall, this quarter laid the foundation for not only automating decision predictions but also making those predictions interpretable and auditable.

Quarter 3: Anomaly Detection and Scoring

After building the classification model, the next step was to identify potentially incorrect approvals that the model might miss. To do this, I implemented unsupervised anomaly detection using Isolation Forest and a Variational Autoencoder (VAE). These models were applied to the approved requests to flag those that looked unusual based on input patterns or had low model confidence.

The VAE was trained on typical approval data and flagged outliers using reconstruction error. Isolation Forest detected anomalies based on structural deviations in the features. I used percentile-based thresholds to avoid hard cutoffs and compared the two models using metrics like KL divergence and score correlation.

This phase resulted in a ranked list of suspicious approvals, which combined anomaly scores with classifier confidence. These outputs were prepared for integration into downstream dashboards to support audit reviews.

Quarter 4: Documentation and Final Presentation

In the final phase, I focused on preparing a comprehensive document that detailed the full project workflow, including the data used, modeling pipeline, anomaly detection logic, and key findings. This document was intended for internal stakeholders to evaluate the system's effectiveness and consider its future integration into audit workflows.

Alongside the report, I also worked on my final internship presentation, which summarised the problem, approach, results, and business impact. This phase ensured the work was clearly communicated and positioned for potential adoption by the team.

III. Key Challenges and Successes

Throughout the internship, I encountered several technical and strategic challenges. Handling imbalanced data particularly the underrepresented "conditionally approved" class required careful model selection and evaluation. Avoiding data leakage during preprocessing and ensuring fair train-test splits were also critical to building reliable models. Tuning the classifier to avoid overfitting while still performing well across all classes was a balancing act. Another key challenge was interpreting and communicating model outputs in a way that non-technical stakeholders could understand and act upon. This became especially important during anomaly detection, where there were no labeled examples to validate against. Translating technical results into meaningful business insights required a continuous feedback loop between model performance and operational relevance.

Despite these challenges, there were several rewarding outcomes. I successfully delivered a working end-to-end system that combined classification and anomaly detection to support automated audits. One of the most promising results was that the system was strong enough to flag a number of potentially incorrect or risky approvals cases. Using SHAP, I was able to generate interpretable explanations for these predictions, which helped build trust in the model's reasoning. I also received positive feedback on my documentation, which demonstrated the value of combining technical depth with clear communication.

IV. Learning and Professional Development

This internship gave me valuable exposure to how machine learning systems are built and applied in real-world, high-stakes environments. I worked with complex, messy data and gained hands-on experience in model development, evaluation, and explainability. Using tools like SHAP, I saw how interpretability is essential when models influence business decisions. I also improved my ability to communicate technical insights clearly through documentation and presentations. Most importantly, the experience reinforced my interest in building reliable, interpretable AI systems that support human decision-making in impactful domains.

Acknowledgment

I would like to sincerely thank my manager, Subha, for her continuous support throughout the internship ensuring I had no blockers, and that both my onboarding and project work progressed smoothly. I'm especially grateful to my mentor, Yuzheng An, who provided invaluable technical guidance throughout my project whether it was brainstorming ideas, helping shape the approach, or guiding me through model development. She also encouraged me to participate in internal workshops and science fair sessions, helping me gain confidence in sharing my work and refining my ideas. And of course, the casual coffee chats along the way made my first industry experience even more enjoyable I truly couldn't have asked for a better mentor.

I also want to extend my thanks to the Security Data Intelligence (SDI) team, especially Aditi, for taking the time to help me understand what an ECD is and walking me through the end-to-end workflow in detail.

Finally, a huge thank you to Catia Silva, my internship program coordinator, and Lisa Hibbs, my academic advisor at the University of Florida, as well as my faculty mentor, Professor Laura Melissa Cruz Castro, for their continued support and guidance throughout the internship experience.