Client Onboarding for Private Banking

Julius Bär Challenge - ETHZ ACE Datathon 2025

Andreas Spanopoulos, Hector Maeso, Marcell Schnieder, Juan Garcia

TL;DR

We built an explainable, modular system that uses handcrafted rules, with some relying on the OpenAI API for formatting natural language fields. For the remaining cases, clients are filtered using a classifier trained on LLM embeddings.

1 System architecture

An overview of the main components of our intelligent onboarding system and how they interact is presented in Figure 1. In the following enumeration we list the individual steps of the process, the components involved, their roles and how they sequentially interact to provide an onboarding decision.

1. Client `.json` data is loaded into a **client data container** and passed to the **onboarding model**.
2. The model checks basic constraints via the **explainable rules knowledge base**. If a rule fails, the **client** is rejected and the violated rule is logged.
3. If not rejected, complex fields are parsed using the **OpenAI API** and validated under the **explainable rules knowledge base** constraints. Failures lead to rejection with the conflictive field being logged.
4. If the client overcomes the **handcrafted rules module** embedding are generated via **Roberta**, then classified by an **MLP** into accept/reject.

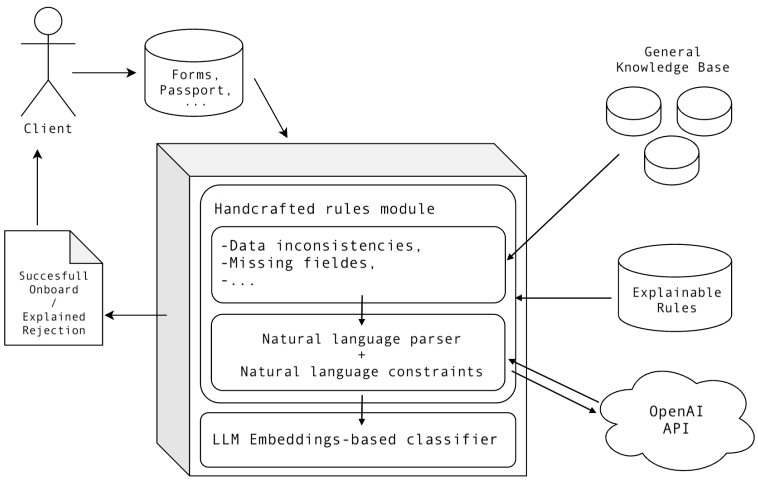

Figure 1: General overview of our intelligent onboarding system

2 Explainability and risk analysis

Our system combines **language models** with **fully explainable rules** to handle rejections. The **majority of inconsistent cases** are addressed through deterministic rules, enabling us to provide clients with **immediate and transparent explanations**. Only a small number of **complex cases**, handled by language models, may require **human oversight** to clarify the rejection rationale.

At the core of our framework are **semantically correct rules by design**, which are **inherently resistant to overfitting**. This guarantees consistency and robustness across decisions.

A key strength of our approach is that **handcrafted rule-based rejections** are **highly reliable**, with nearly all such cases being **accurately annotated**. Even when considering the entire system, the **false rejection rate remains below 1%**, significantly reducing the risk of **reputational harm** from incorrect decisions.

Future improvements should focus on minimizing **undesired client acceptances**, which are more likely to trigger **regulatory concerns**.

3 Resource consumption and scalability

We consider **resource consumption** to be relevant, as it impacts both the **operational cost** for the company and the **environmental footprint** of the process. In this regard, a single client is classified by the system in approximately ~ 1 second on a **GPU-equipped laptop** with internet access (required for communication with the **OpenAI API**). The **API cost per client** is below $0.01 CHF.

As shown in the system architecture, rules are executed in order of **increasing resource consumption**, so that more advanced and expensive components are only triggered for clients that are **harder to classify**. The **development cost** is also minimal — less than $10 CHF were spent on OpenAI API usage during the entire development phase.

Additionally, the **modular structure** of our architecture facilitates easy **integration of new rules** or **replacement of components**, such as introducing more **efficient or powerful language models** as needed.

4 Results

The proposed onboarding agent achieves a validation accuracy of $\sim 95\%$. We highlight that this accuracy has the potential to be further improved by talking with onboarding experts or incorporating human assistance between submodules of the agent. Further insights on the model output are provided in the slides accompanying this file.

5 Conclusions

Our system achieves a very high degree of **explainability** for most rejections, while allowing the use of **language model-based rules** for the few more complex cases. It is designed to progressively reject clients through **multiple stages**, ensuring very high confidence in the rejection decisions.

The core of the framework relies on **semantically correct handcrafted rules**, which by construction **do not overfit**. A test accuracy of **94%** is achieved, with fewer than **10% of the errors** being true rejections mistakenly flagged as acceptances.

As previously discussed, the framework is well-suited to managing client relations effectively—**minimizing false rejections** and providing clear explanations for most flagged cases. At the same time, additional layers should be implemented to better handle **incorrectly accepted clients**, primarily to mitigate potential issues with **regulatory compliance**.