Explainable ML Pipelines with Agentic AI in Healthcare and Finance

Group 11

Background and Motivation

Traditional ML pipelines require significant manual effort: cleaning datasets, engineering features, tuning hyperparameters, and evaluating models. These steps are repetitive, time-consuming, and often opaque. AutoML platforms attempt to automate parts of the process but function as black boxes, providing little transparency into the decisions being made.

With the rise of large language models (LLMs), a new paradigm is possible: Agentic AI system, where specialized agents collaborate to handle different stages of an ML pipeline. Instead of one monolithic algorithm, multiple agents can take on roles such as data profiling, feature suggestion, model tuning, evaluation, and risk assessment. Guardrails and human-in-the-loop (HITL) oversight ensure accountability and traceability.

This project leverages this paradigm to build a lightweight, reproducible ML pipeline driven by modular agent instances, focusing on structured datasets in healthcare and finance, domains where reliability and explainability are critical.

Problem Statement

We aim to design and implement an **agent-driven ML workflow system** that:

- Works with tabular CSV datasets from healthcare (disease prediction) and finance (fraud detection).
- Uses LLM-based agents to analyze, propose, and execute pipeline steps.
- Demonstrates judge agents that provide structured feedback and guide retraining loops.
- Runs feasibly on local hardware with reproducible results.

Importance of the Problem

- **Healthcare**: Disease risk prediction models can support decision-making but often require extensive preprocessing and validation. Agents can automate these tasks while providing documented reasoning.
- **Finance**: Fraud detection requires constant retraining and monitoring. Agents can iteratively refine models while judge agents highlight risks of overfitting or drift.
- **General impact**: By moving beyond black-box AutoML, our framework shows how LLMs can make ML pipelines **transparent, iterative, and auditable.**

Challenges

- **Local execution**: Building agent loops that run efficiently on personal hardware (e.g., MacBook M2).
- **Dynamic iteration**: Designing feedback-driven loops that improve models without infinite cycling.
- **Consistency**: Ensuring that agents' outputs remain reproducible and not random.
- **Baseline comparison**: Evaluating against AutoML systems to demonstrate added value.

Proposed Solution Approach

Our architecture is designed as a **multi-agent workflow system**, where each agent is instantiated separately and assigned a specialized role. Instead of dividing into large "crews," the system coordinates **multiple task-specific agents** that interact dynamically:

- **EDA Agent**: Reviews dataset schema, statistics, and missing values.
- **Feature Engineering Agent**: Suggests and applies transformations.
- **Hyperparameter Tuning Agent**: Proposes model parameter adjustments, triggers retraining executors, and evaluates performance.
- **Model Selection / Judge Agent**: Reviews candidate models, critiques performance, and recommends iterations or approval.

These agents may run as **different LLM instances** (e.g., DeepSeek R1 7B locally for feature engineering, another instance for hyperparameter tuning, another for judging). This modularity allows scalability and avoids coupling roles into fixed crews.

Novelty: Unlike AutoML, which produces a single "best" model and is cloud-based, our system is local, lightweight, and builds a **feedback-driven loop** where specialized agents critique, refine, and retrain models iteratively. Human-in-the-loop (HITL) checkpoints further ensure traceability and explainability.

Deliverables

- Scripts for dataset profiling, feature engineering, model training, and evaluation.
- A **lightweight multi-agent orchestration layer** that coordinates specialized LLM instances (EDA, feature engineering, hyperparameter tuning, judging), designed to run on local hardware.
- Executors to trigger retraining and evaluation, with feedback loops for iterative refinement.
- End-to-end reproducible demos on:
    - **Healthcare dataset** (e.g., diabetes/heart disease prediction).
    - **Finance dataset** (e.g., credit card fraud detection, spam detection).
- A **qualitative comparison with AutoML baselines**, emphasizing workflow transparency, iterative refinement, explainability, and lightweight local feasibility rather than raw accuracy.

Timeline

- Week 1–2: Literature review, define agent roles, prepare datasets.
- Week 3–4: Implement Modeling Crew with executors (EDA → training → evaluation).
- Week 5: Add Judge/Risk Crew and feedback-driven retraining loops.
- Week 6: Experiments on healthcare and finance datasets; compare with AutoML.
- Week 7: Documentation, reproducibility testing, final demo prep.

Division of Work

- Samarth Batra: Dataset profiling, feature engineering scripts.
- Meet Zalavadiya: Agent role design, planner/executor implementation.
- Akshat Bishnoi: Model training/evaluation pipeline, AutoML baseline comparisons.
- Sarthak Singh: Judge/Risk Crew, HITL checkpoints, feedback logic.
- Omkar Rane: Documentation, reproducibility framework, final testing.