Agentic HR Churn Risk Dashboard

By Naman Soni
Email: naman21csu438@ncuindia.edu
Mobile: 9166157517

Project Overview

This project is an end-to-end HR analytics and intervention system that:

- Predicts employee churn risk using machine learning.
- Uses a Large Language Model (Claude via LangChain) to generate human-readable risk assessments and recommended HR actions.
- Provides a Streamlit dashboard for HR to review high-risk cases and auto-draft empathetic emails for intervention.

Workflow Diagram

```
graph TD
    A[Employee Data (CSV)] --> B[ML Model (Jupyter Notebook)]
    B --> C[Churn Predictions (JSON)]
    C --> D[LLM Risk Assessment (Jupyter Notebook)]
    D --> E[LLM Decisions (JSON)]
    E --> F[Streamlit Dashboard]
    F --> G[HR Email Generation]
```

Detailed Workflow Steps

Step 1: Data Science & Prediction

- Clean and process employee data in `Main_Notebook.ipynb`.
- Train and evaluate a churn prediction model (Random Forest, Gradient Boosting, etc.).
- Export predictions to `churn_predictions.json`.

Step 2: LLM Risk Assessment

- Use `Agent(llm_recommender).ipynb` to convert predictions into structured, human-readable risk assessments and recommended actions using Claude.
- Output is saved as `churn_llm_decisions.json`.

Step 3: HR Dashboard & Email Generation

- Run `agentic_dashboard_py.py` with Streamlit.
- Upload `churn_llm_decisions.json`.
- Select high-risk employees and auto-draft professional, empathetic HR emails for intervention.

Key Files & Their Roles

- **data/employee_churn.csv**: Main employee dataset.
- **Main_Notebook.ipynb**: Data cleaning, feature engineering, model training, and prediction.
- **churn_predictions.json**: Output of the ML model, used as input for the LLM.
- **Agent(llm_recommender).ipynb**: Uses Claude LLM to generate risk assessments and recommendations.
- **churn_llm_decisions.json**: Output of the LLM, used as input for the dashboard.
- **agentic_dashboard_py.py**: Streamlit dashboard for HR to review high-risk cases and generate emails.

Sample Data & Outputs

Sample from `employee_churn.csv`:

avg_monthly_hrs	department	filed_complaint	last_evaluation	n_projects	recently_promoted	salary	satisfaction	status	tenure
221	engineering	0.93	4			low	0.82	Left	5

Sample from `churn_predictions.json`:

```
{
  "prediction": "Left",
  "churn_probability": 1.0,
  "features": {
    "avg_monthly_hrs": 135,
    "filed_complaint": 0.0,
    ...
  }
}
```

Sample from `churn_llm_decisions.json`:

```
{
  "employee_id": 0,
  "status": "HIGH RISK",
  "reasons": ["Low satisfaction score (0.45)", ...],
  "recommended_actions": ["Schedule immediate manager check-in...", ...],
  "summary": "Employee shows multiple risk factors including low satisfaction..."
}
```

How to Run the Project

1. Install dependencies

```
pip install streamlit langchain-anthropic langchain-core
```

2. (Optional) Set up a virtual environment

```
python -m venv .venv
source .venv/bin/activate  # On Windows: .venv\Scripts\activate
```

3. Get a Claude API key

- Sign up at Anthropic (https://www.anthropic.com/) and obtain your API key.

4. Run the Dashboard

```
streamlit run agentic_dashboard_py.py
```

- Open the local URL shown in your terminal (usually http://localhost:8501).
- Enter your Claude API key in the sidebar.
- Upload `churn_llm_decisions.json`.
- Select employees and draft/send emails.

Contact Information

- **Name:** Naman Soni
- **Email:** naman21csu438@ncuindia.edu
- **Mobile:** 9166157517
