Task: Applied AI Intern Task for Internship at Swiggy c/o Ahijeet Pushkarna

Context: You are working for a food review platform that wants to improve insights for restaurant partners. The platform has given you a dataset of customer reviews and restaurant-level information.

Your job is to build an AI-powered solution that can:

1. Summarize customer reviews into short insights.

2. Predict which restaurants are at "High Risk" (likely to face complaints or closure).

3. Deploy the solution as a simple web app.

Dataset Provided: Restaurant_reviews_dataset.csv

Columns:

- review_id → Unique review identifier
- restaurant_id → Restaurant identifier
- date → Review date
- rating → Customer rating (1–5)
- review_text → Customer's written feedback
- monthly_sales → Monthly revenue of the restaurant (₹)
- complaints_count → Complaints in the last month
- avg_rating → Average customer rating for the restaurant
- staff_turnover → Staff turnover (last 6 months)
- risk_label → Target variable (1 = High Risk, 0 = Low Risk)

Your Task

1. Text Summarization (NLP): Build a model that summarizes all reviews of a given restaurant into 2–3 meaningful sentences. Use a transformer-based model (e.g., BART, T5) or any other summarization technique. Evaluate the summaries with ROUGE and one semantic metric (e.g., cosine similarity or BERTScore).

2. Risk Prediction (ML Classification): Build a model to predict the risk_label of a restaurant. Use both restaurant-level features (sales, complaints, avg rating, turnover) and review-derived features (sentiment, embeddings). Train at least 3 different models (e.g., Logistic Regression, XGBoost, SVM) and compare them. Report metrics: precision, recall, ROC-AUC, F1-score.

3. Explainability: Use SHAP or LIME to explain which factors drive the risk prediction. Provide at least 2 restaurant case studies with explanations.

4. Deployment (Streamlit App): Build a simple web app where a user can:

Enter/select a restaurant ID.

See a summary of its reviews.

See the risk prediction (High Risk / Low Risk).

See the top 3 explanatory factors behind the prediction.

5. (Bonus – Optional)

Add a simple drift detection mechanism:

If new reviews contain unusual words/phrases not seen in training, flag it.

Deliverables

- A GitHub repo with:
- Code + dataset + README.
- Your model training notebooks.
- A working Streamlit app (share a link or video demo).
- A short report (2–3 pages) covering:
- Data preprocessing.
- Modelling choices.
- Evaluation results.

Key insights: Challenges + improvements you would make.

Evaluation Criteria (100 points)

- Data Preprocessing & Feature Engineering – 15
- Summarization Model & Evaluation – 20
- Risk Prediction Models & Results – 20
- Explainability (SHAP/LIME, clarity of insights) – 10
- Streamlit App (functionality & usability) – 20
- Documentation & Report – 10
- Bonus (Drift Detection) – 5

Time Stamps:

Task Given: Sunday, 31Aug, 2025 17:00h

Submission Deadline: Tuesday, 02 Sept, 2025 on or before 17:00h

The goal is not just code, but clear reasoning and communication

Good luck!