Virtual Failure Analysis Project

Internship Project First Presentation

Zeng Fanyi A0285753R

Data Scientist Intern

IT BIA CCA

Table of contents

Business Objective

Background

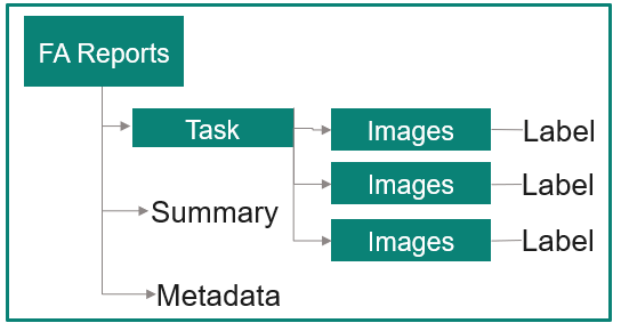
During the microchip Failure Analysis (FA) , to identify the Physical Root Cause historical reports can bring 3 value adds for customers:

- Failure Mode
- Failure Mechanism
- Benchmark

Currently, **not all FA reports and images are 100% labeled** for reference, and **most are labeled manually**

Objective

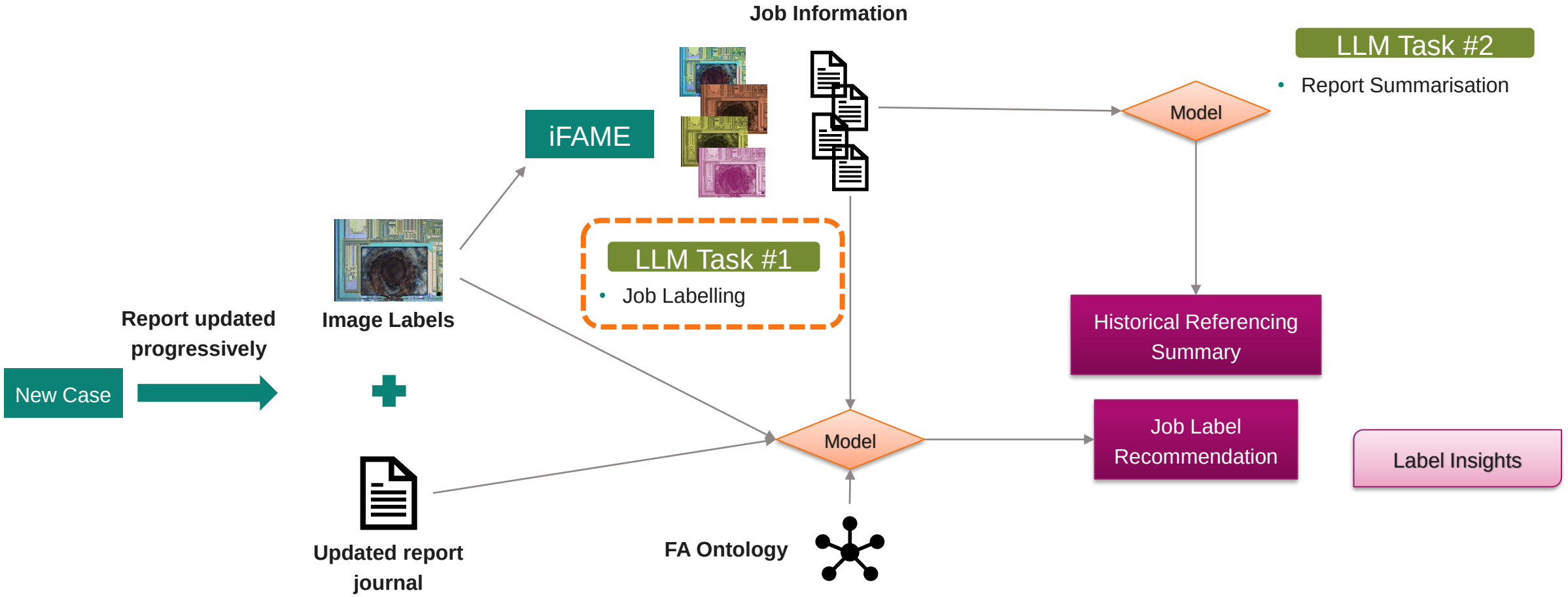
As IT support department, we want to use **LLMs** to assist on **Labelling Historical Reports** & **New Reports** based on the FA Ontology of Failure Modes & Textual Data from the FA Reports



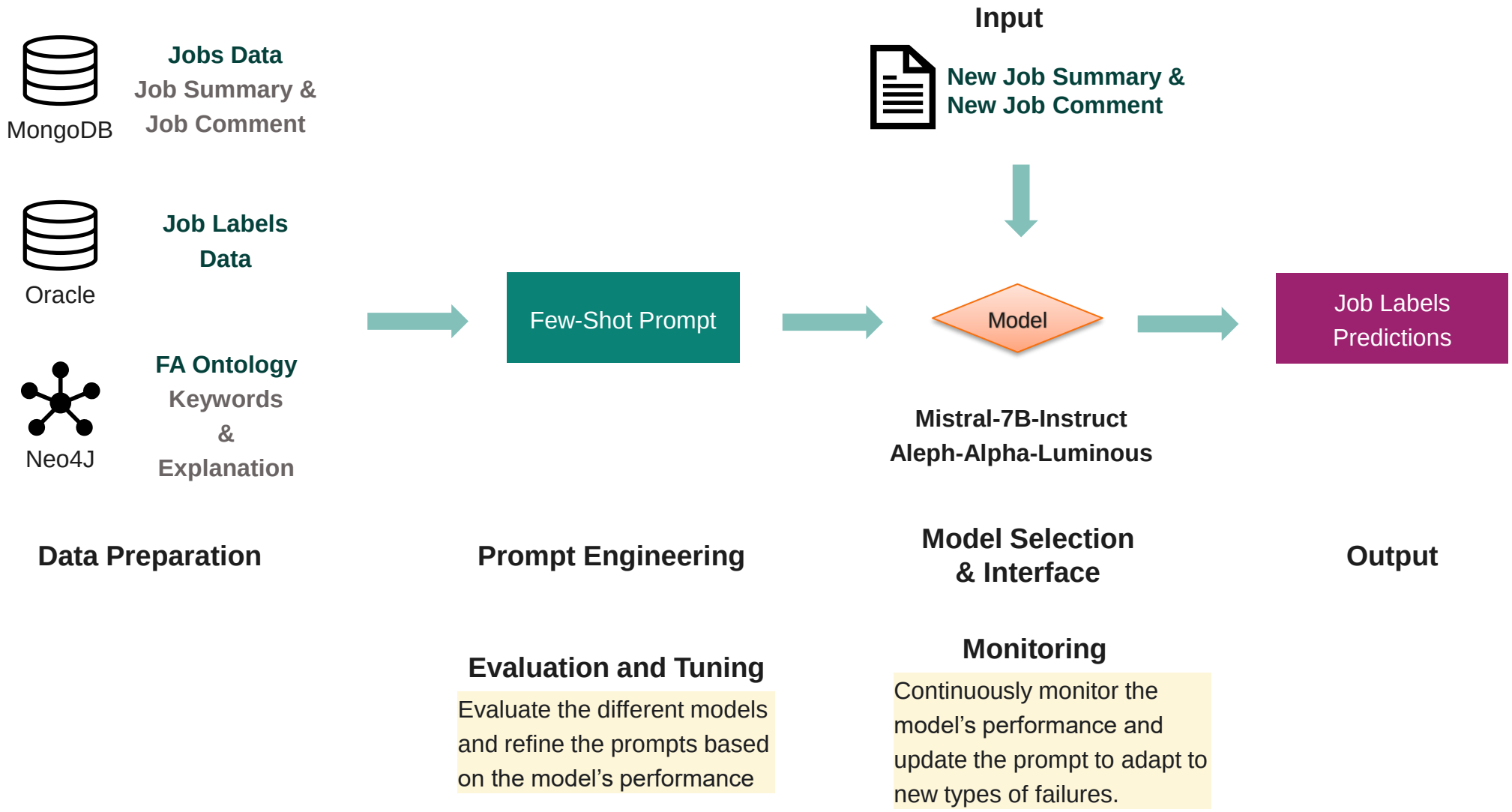
End Users

Infineon FA Engineers

Workflow - Failure Analysis



Workflow - Job Labelling



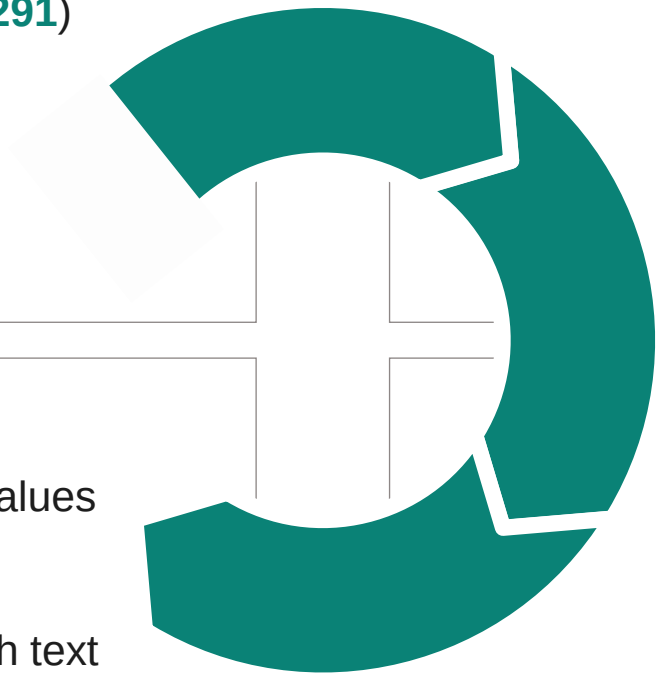
Data Preparation

1. Data Loading

- Job Data: Extract from MongoDB (**980,476**)
- Job Details Data: Extract from Oracle (**27,291**)
- FA Ontology: Extract from Neo4j (**55**)

2. Data Merging and Filtering

- Merge Labelled Data & Job Details Data
- Filter dataset with presence of electrical fault



4. Data Quality Checks

- Empty Value Rate: Proportion of missing values
- Token Length: Average number of tokens
- Language Detection: Percentage of English text

3. Data Preprocessing

- Text cleaning:
- Language filtering
- Label Splitting

Preprocessed Training Data: **26,646**

Prompt Engineering – Structure & Example

Define the multiple-label prediction task

Provide all descriptions of failure modes (FA Ontology)

Provide the guidelines for model outputs

Provide few annotated examples to illustrate how the model should analyse the failure modes

Insert the job summary and comment to be analysed, the model will predict the content after 'Failure:'

```
[INST] You are a classifier, and your task is to analyze the given 'Comment' and assign the appropriate
'Failure(s)' from the provided list:

<<<
- Label: Hardware Failure
  Keywords: device malfunction, broken, not working
  Comment: Issues related to physical components not functioning as intended.
- Label: Software Bug
  Keywords: error message, crash, glitch
  Comment: Problems caused by errors in the software code.
>>>

To complete your task, follow these guidelines:
1. Carefully examine the 'Comment' provided. Assign the relevant 'Label(s)' based on your analysis.
2. Identify keywords or phrases that match the 'Keywords' and 'Comment' descriptions associated with each label.
3. Output ONLY the most relevant 'Label(s)' from the list, separated by semicolons (;). Do NOT include any
explanations or other text.

### Here are some examples to guide your labeling process:

Example 1:
Comment: My phone screen suddenly went black and won't turn on.
Failure: Hardware Failure

Example 2:
Comment: The app keeps crashing whenever I try to open a document.
Failure: Software Bug
……

### Now, please label the following comment:
Comment: {input}
Failure:
[/INST]
```

Prompt Engineering – Similarity-Based Few-shot Learning

Goal: Improve model performance during inference using a few relevant examples.

Method:

- Compute TF-IDF Matrix: Represent the importance of words in the entire dataset using Term Frequency-Inverse Document Frequency.
- Calculate Cosine Similarity: Measure the similarity between the input text and each document in the dataset.
- Select Top Samples: Identify the most similar samples (with the highest cosine similarity scores) to be used as few-shot examples.

Model Selection – Before Similarity-Based Few-shot

Models: Mistral and Aleph Alpha

Mistral: Mistral-7B-Instruct

- Longer of prompt length
  - The model has maximum context length of 32k.
- Results of 500 unseen Data (Manually Select)

  Total predictions: 500

  Error predictions: 0

  Perfect match rate (all): 8.6%

  Partial match rate (all): 68.8%

  Perfect match rate (valid): 8.6%

  Partial match rate (valid): 68.8%

Aleph Alpha: luminous-supreme-control-20240215

- Limitations of prompt length
  - The model has maximum context length of 2048.
- Results of 500 unseen Data (Manually Select)

  Total predictions: 500

  Error predictions: 124

  Perfect match rate (all): 6.04%

  Partial match rate (all): 27.57%

  Perfect match rate (valid): 9.31%

  Partial match rate (valid): 29.79%

Model Selection – After Similarity-Based Few-shot

Mistral: Mistral-7B-Instruct

Manually Select

- Results of 500 unseen Data
  - Total predictions: 500
  - Error predictions: 0
  - Perfect match rate (all): 8.6%
  - Partial match rate (all): 68.8%
  - Perfect match rate (valid): 8.6%
  - Partial match rate (valid): 68.8%

Similarity-Based

- Results of 500 unseen Data
  - Total predictions: 500
  - Error predictions: 0
  - Perfect match rate (all): 34.5%
  - Partial match rate (all): 64.0%
  - Perfect match rate (valid): 34.5%
  - Partial match rate (valid): 64.0%

Current Issues and Future Plan

1. **Explore newest model version**
2. **Refine Prompt**
3. **Filter the failure type / Add constrains**
    - Using the keywords to filter the failure type
    - Only provide the context about the possible Failure modes
4. **Generate the reasonable description from the existing data**
    - Job summary + Job comment + existing label -> explanation of ontology

Other Ongoing Tasks: QM Chatbot Development Support (LangChain, text 2 sql), several AIML desk research.


Infineon