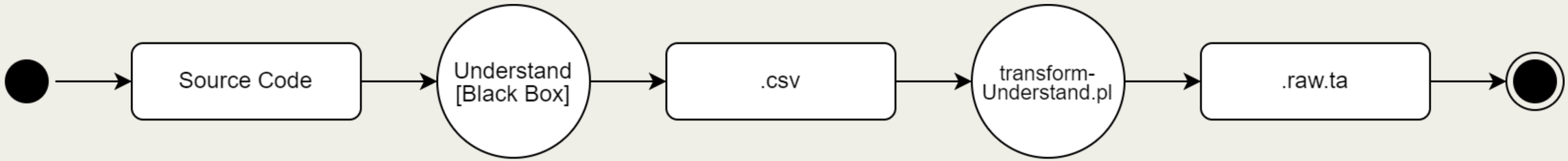
Apache Flink Dependency Extraction

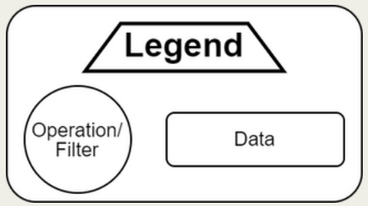
TEAM FLINKFORCE

AGENDA

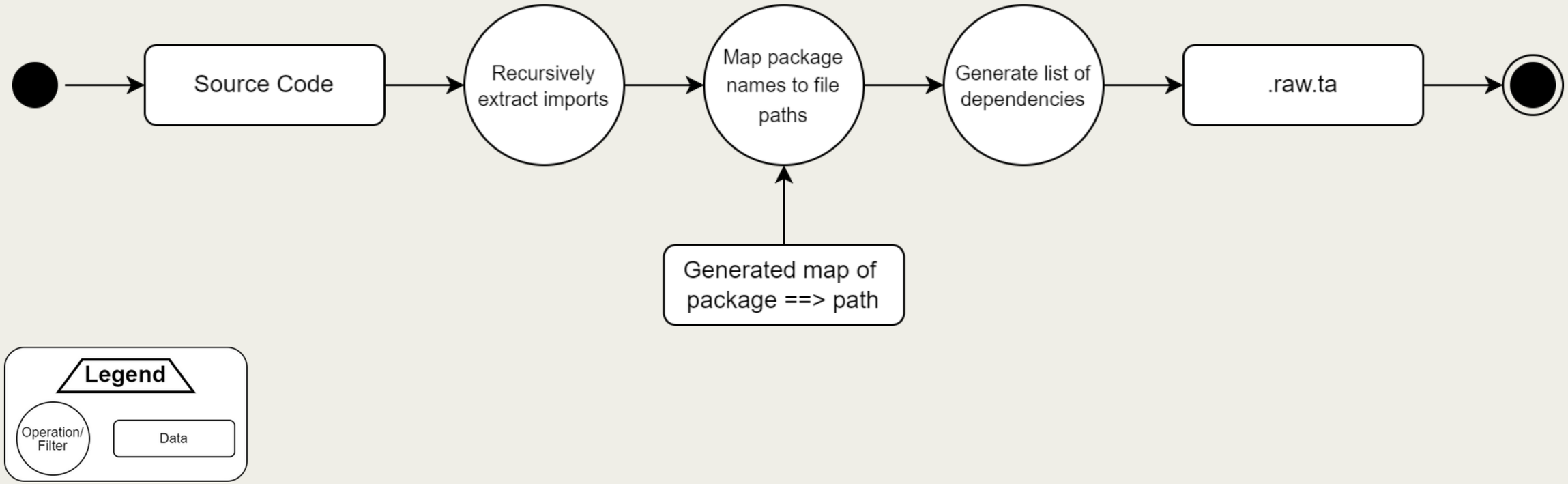
- Dependency Extraction Techniques Overview
  - Technique #1: Understand
  - Technique #2: Import Checking
  - Technique #3: LLM
- Comparison Process
- Stratified Sampling Process
- Quantitative Analysis
- Qualitative Analysis
- Limitations & Lessons Learned
- Live Demo
- Q&A

TECHNIQUE #1: UNDERSTAND

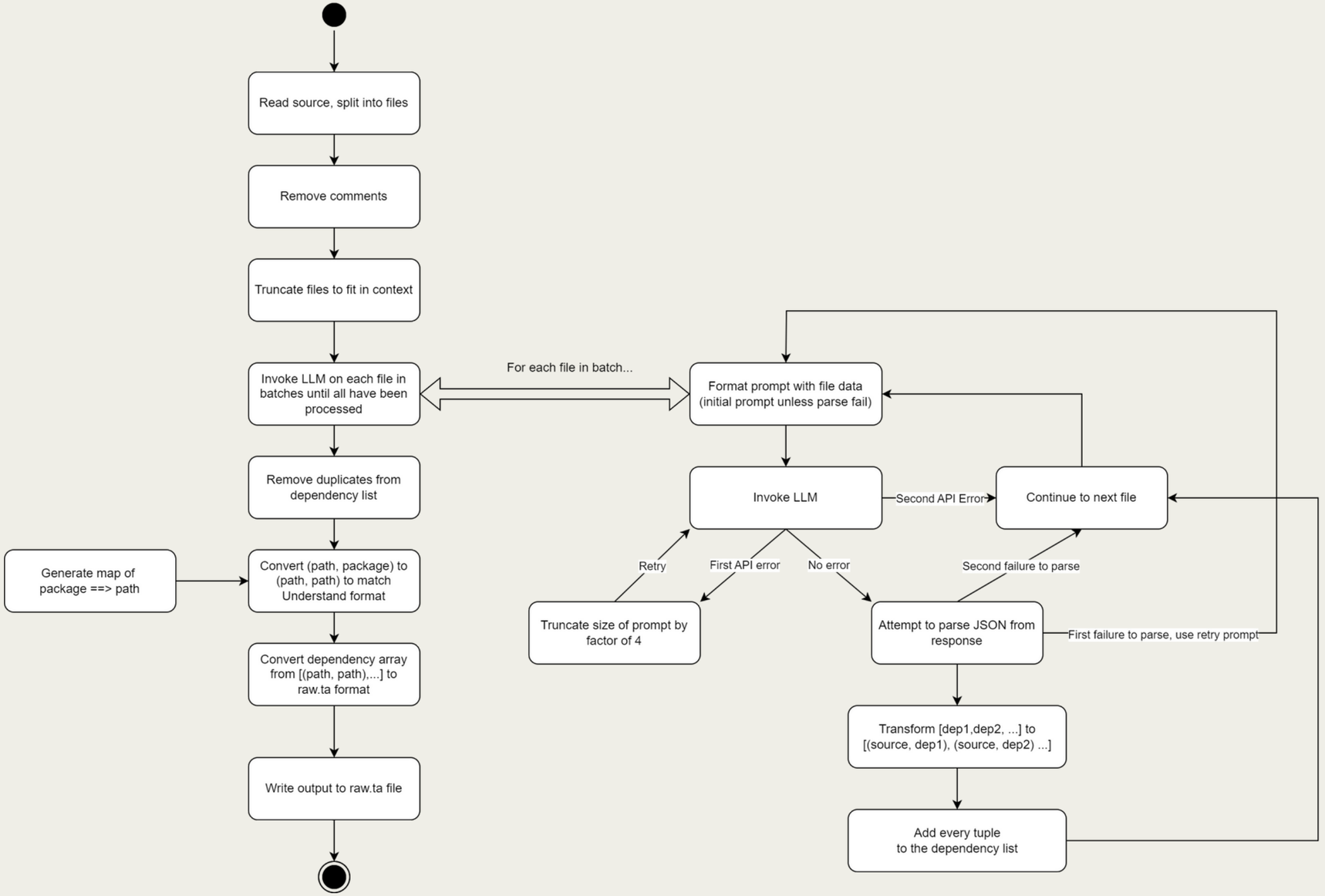




TECHNIQUE #2: IMPORT CHECKING



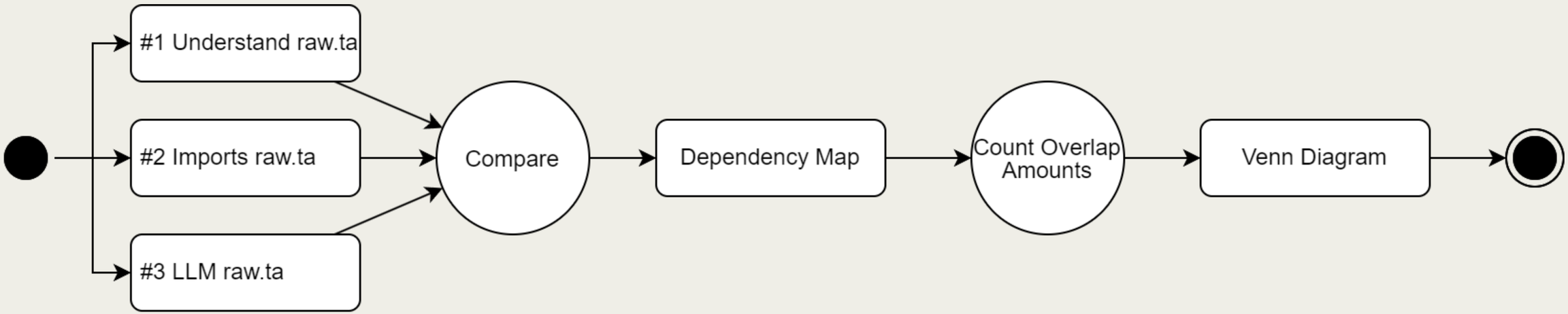
TECHNIQUE #3: LLM

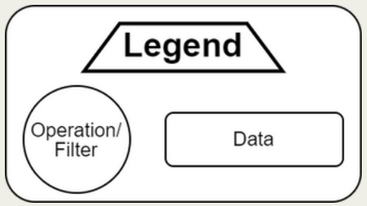


Prompt:

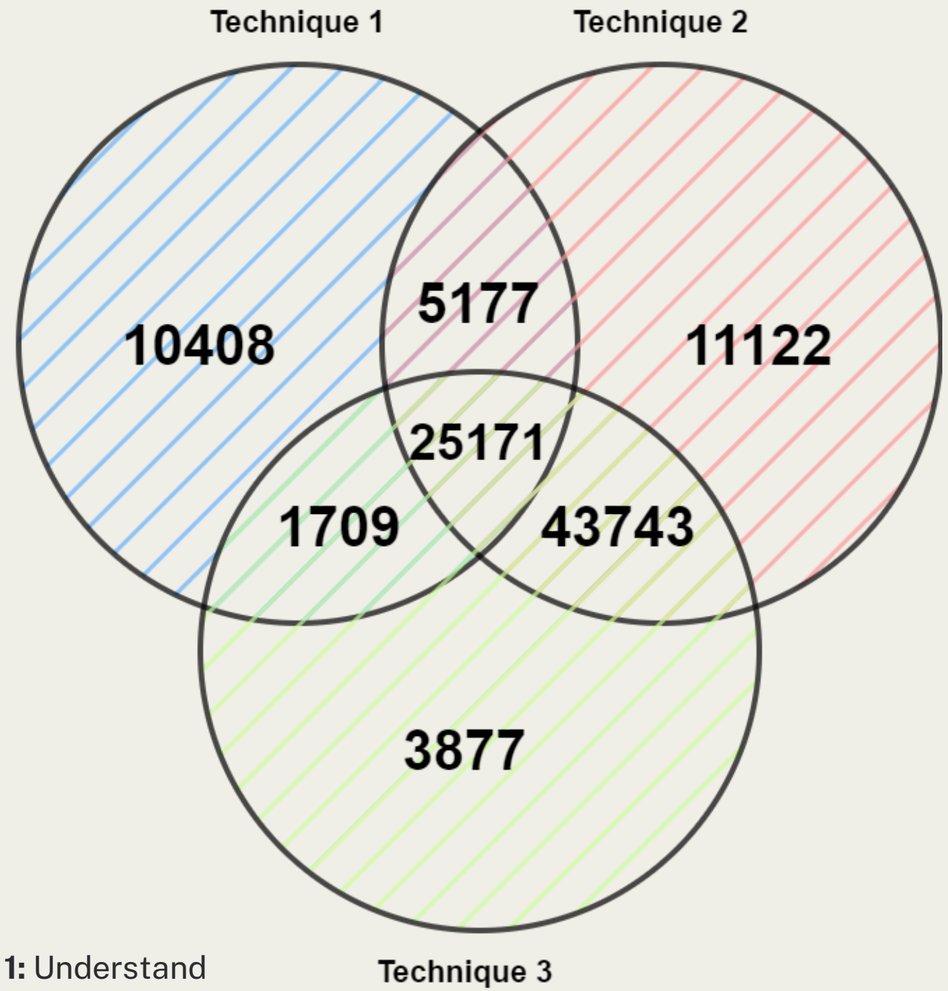
“Instruction: You will only return valid JSON. Given the following code, extract any internal dependencies. Output must be a valid JSON array of strings. For the given code you must determine what external files or packages it depends on, and return them.
File Path: {file_path}
Code: {source}
Answer in valid JSON: \n\n###\n\n"

COMPARISON PROCESS





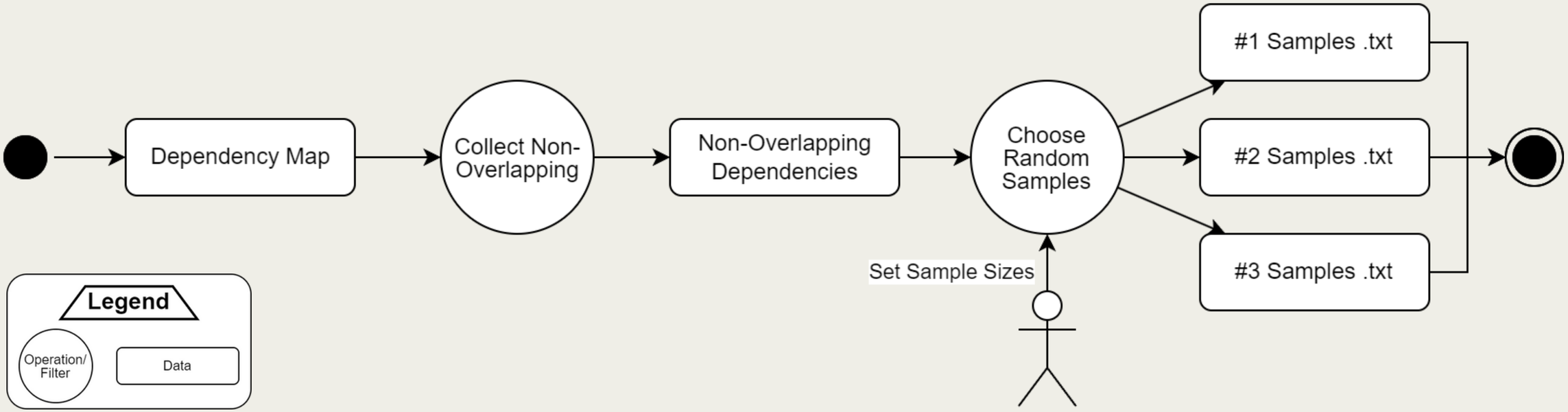
QUANTITATIVE RESULTS OVERVIEW



Technique 1: Understand
Technique 2: Import checking
Technique 3: LLM

Section	Amount
Tech_1	10 408
Tech_2	11 122
Tech_3	3 877
Tech_1,2	5 177
Tech_1,3	1 709
Tech_2,3	43 743
Tech_1,2,3	25 171

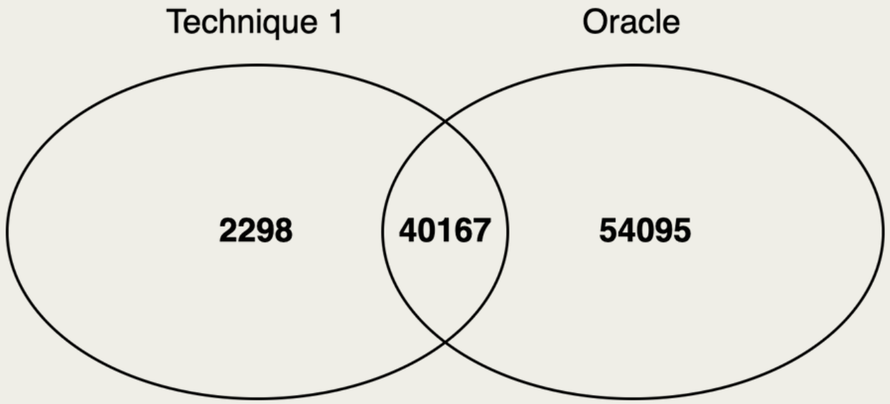
STRATIFIED SAMPLING PROCESS



Stratified Sampling Breakdown		378 cases
T1%	40.97%	155 cases
T2%	43.78%	165 cases
T3%	15.26%	58 cases

Dependency Correctness Percentages	
T1%Actual	77.92%
T2%Actual	94.58%
T3%Actual	87.93%

QUANTITATIVE COMPARISON RESULTS

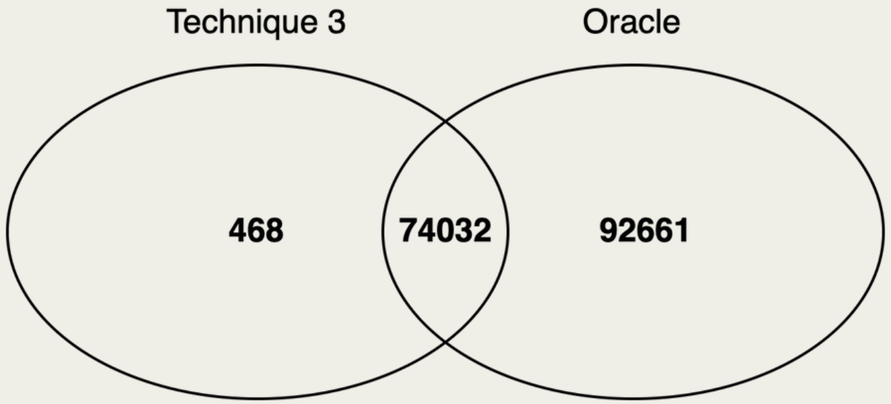


Technique 2

Import Checking

Precision: **99.29%**

Recall: **46.81%**

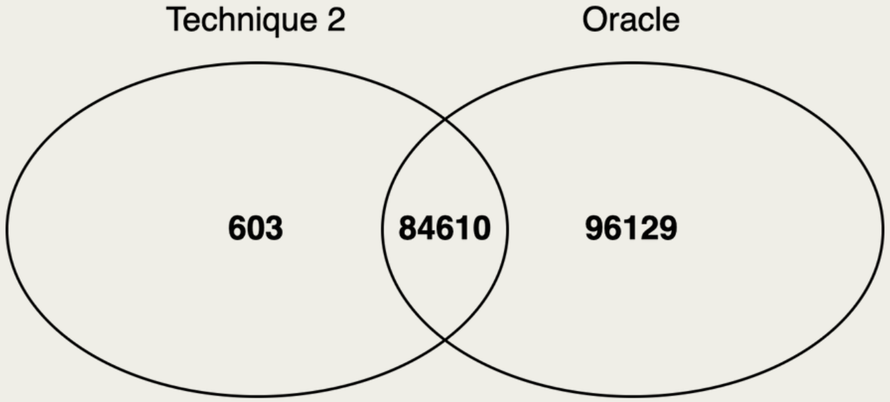


Technique 1

Understand

Precision: **94.59%**

Recall: **42.61%**



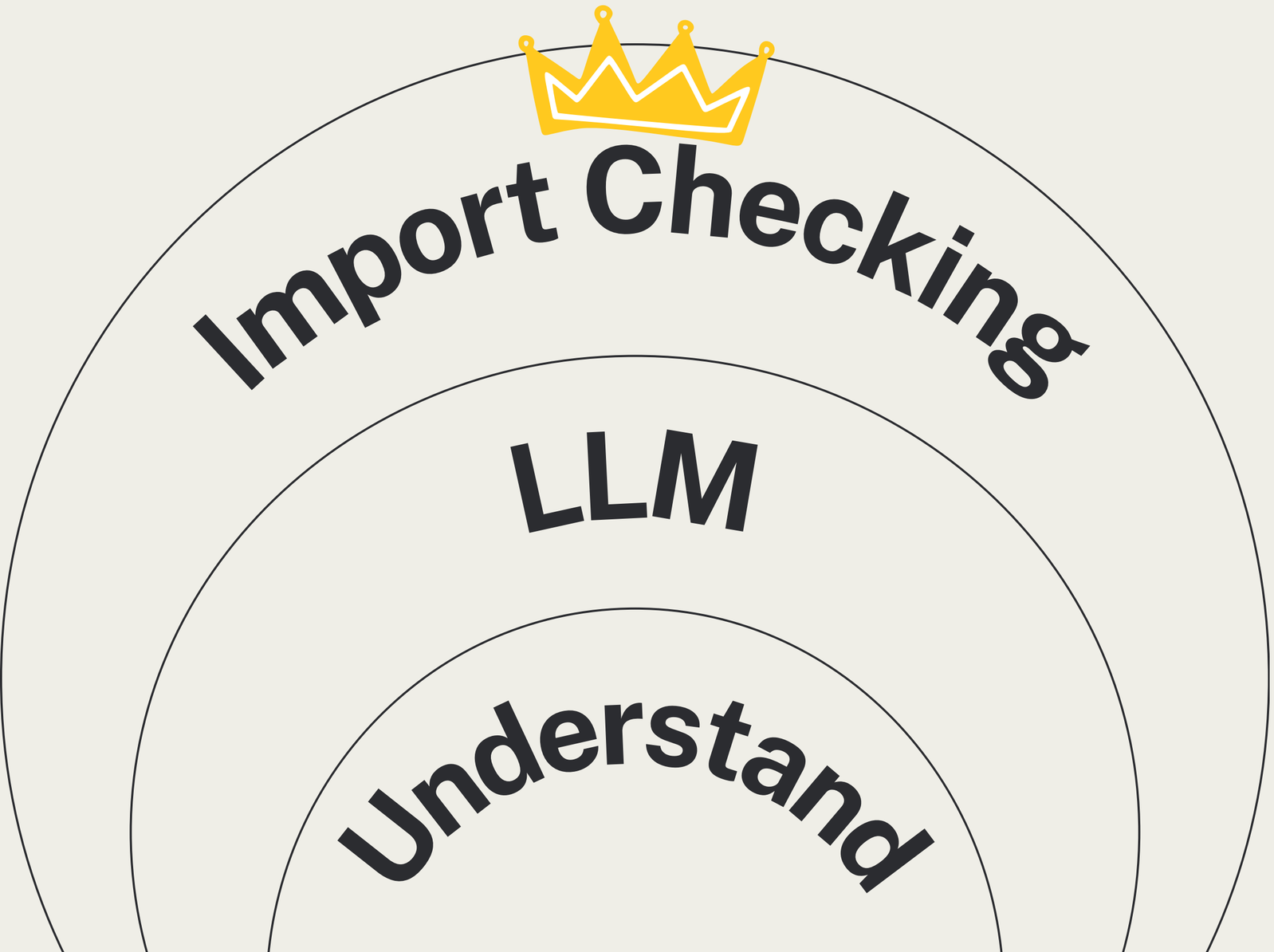
Technique 3

LLM

Precision: **99.37%**

Recall: **44.41%**

QUALITATIVE RESULTS



Import Checking

Highest number of reported dependencies and 99% precision (tied with LLM method)

LLM

Lowest number of reported dependencies and 99% precision (tied with import checking)

Understand

Second highest number of dependencies given and lowest precision & recall values

Import Checking is the best when it comes to precision and recall, however there are other factors to consider

RATIONAL BEHIND THE DIFFERENCES

- Understand is likely **not just looking** at the import statements and has **other** tokens it examines, it is likely eliminating several candidate dependencies and then **prunes** the results, hence the **smallest number of reported dependencies**
- The LLM and Import Check are likely doing **convergent** steps. The LLM is likely looking at **imports** as a **starting point** and then analyzing **other tokens**
- The import check is tailored with a **java** project in mind and works very well for java directories, it may need **heavy modification** for other languages
- The LLM is likely doing pruning of results just like Understand → witnessed with early stopping during tweaking of the model
- LLM may also be looking at the code dynamically (i.e., trace dataflow, variable updates, etc.)

RISKS AND LIMITATIONS OF EACH APPROACH

Understand

Proprietary/closed-source code

Not free to use, must pay or use trials

Cannot be customized, must be used as a black-box

Import Checking

Custom script that may only work well on Flink project directory structure or similar structures.

May struggle if source code is built up of multiple languages

Only looks at the import line, so may miss other dependencies (pom.xml, within package)

LLM

Explainability of results is a key critique of this approach

May rack up high costs if not paying attention

Obtaining OpenAI keys is currently on hold so may need to use other embeddings or wait to fully replicate this for new user

LESSONS LEARNED

Understand

- Free availability is crucial; if not consider better and more modern options
- Fast and lightweight performance
- Decent results achieved

Import Checking

- Allows quick view of dependencies if a project is primarily in Java
- Requires extensive testing and several modifications for other languages

LLM

- Can become very expensive (learned the hard way)
- Requires processing input to optimize cost
- A wide “**prompt space**” may be needed to understand the wide variety of possible results

Live Demo

Thank you!

Q & A