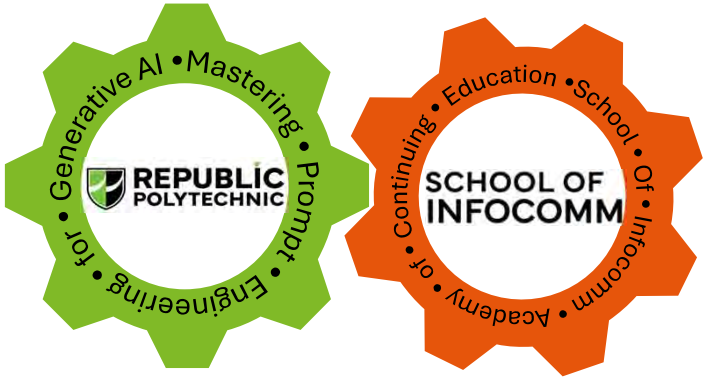
2025

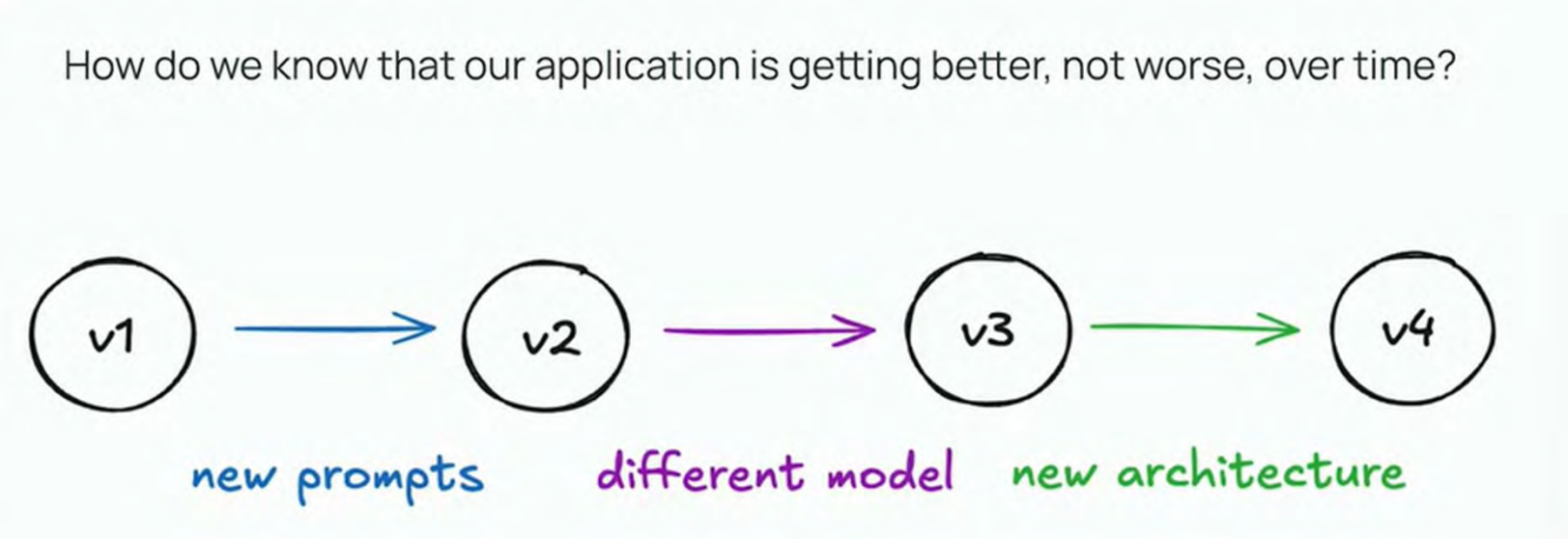


Lesson 16 – Testing and evaluation

Testing and evaluation

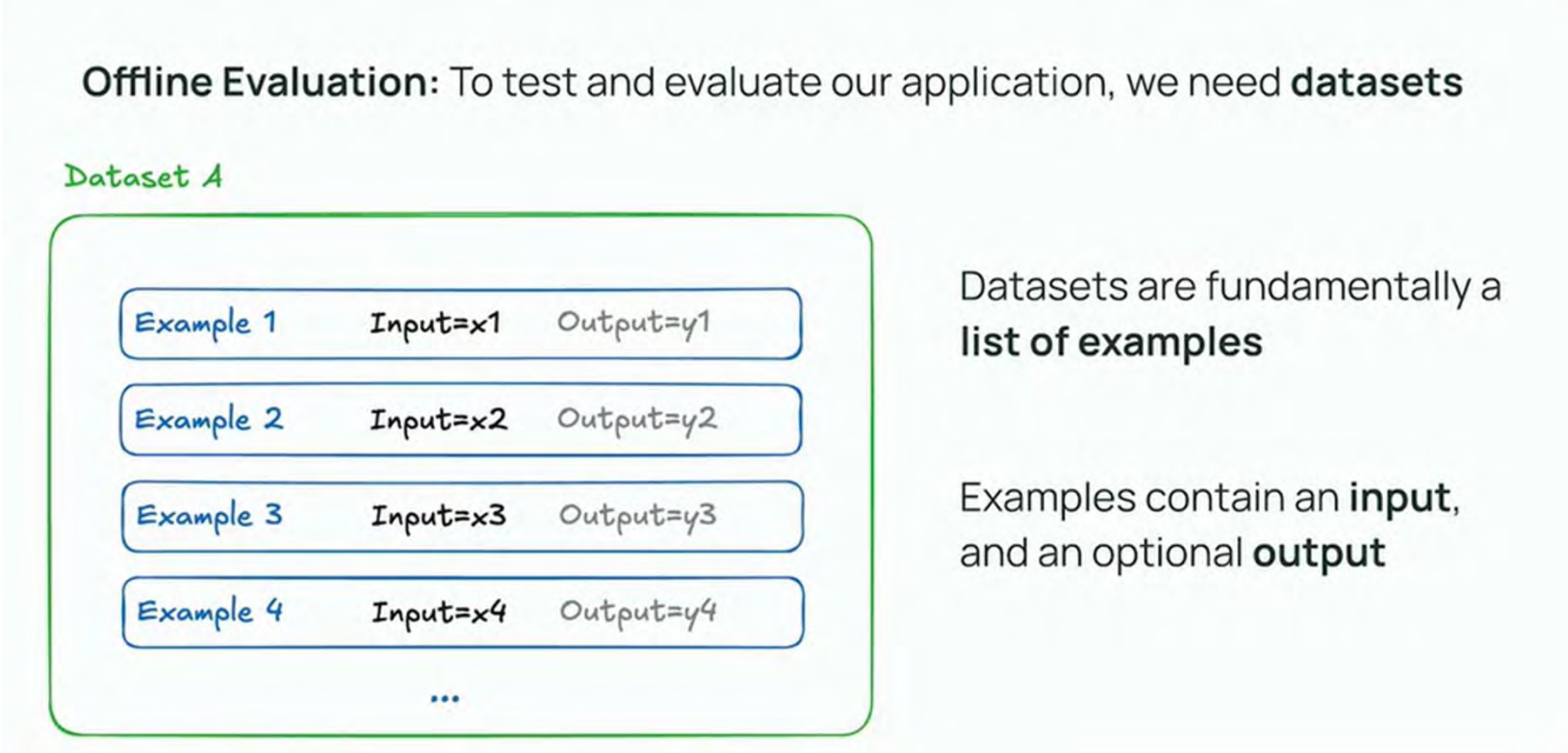
- Create **dataset** in LangSmith and their role in testing LLM applications
- How to **maintain and expand** your datasets over time
- Create **Evaluator** and run **Experiments** over our Dataset
- Inspect Experiment results (**Analyzing** both quantitative and qualitative metrics to assess our app performance)

Datasets



Need to do more than run a few "gut check" tests

Datasets



Activity – Datasets

01 **Navigate to UI and create a dataset**
Dataset name: RAG Application Golden Dataset

02 **Use notebook to add examples**

03 **Tag a version to the initial commit**

04 **Add another trace**
we can add examples directly from trace

05 **Add example to dataset**

Pick one of the exercises in Lesson 13 (Harry Porter or wiki)
Prepare your Golden dataset for that exercise.

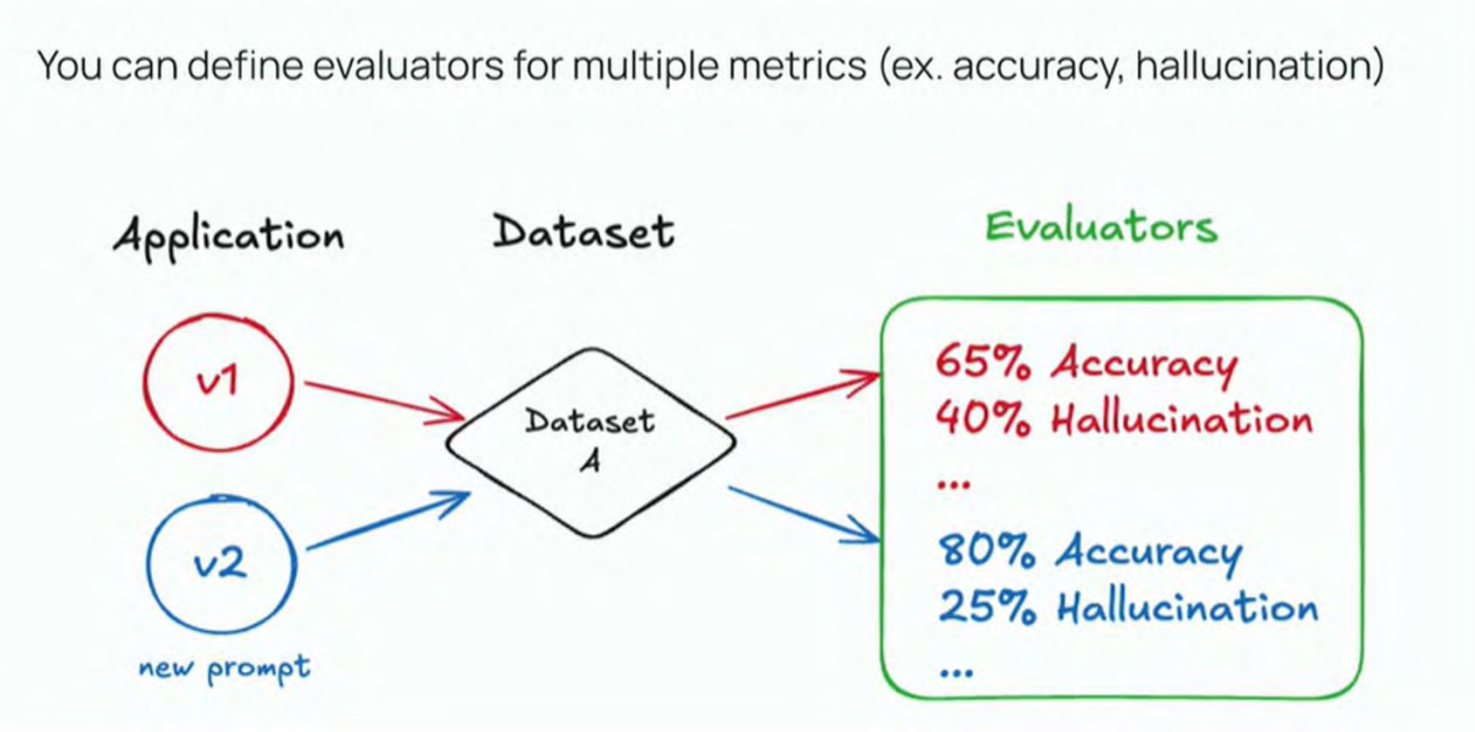
10 mins

Share your dataset url to the padlet

- You can create **separate** datasets for different sub-components
- Create **input schema** to have validation, transformation and syntax highlighting (e.g. input and output)
- **Manual** add
- **AI-generated** examples
- Split datasets
- Share, download, clone dataset

Target to finish by XX:XX

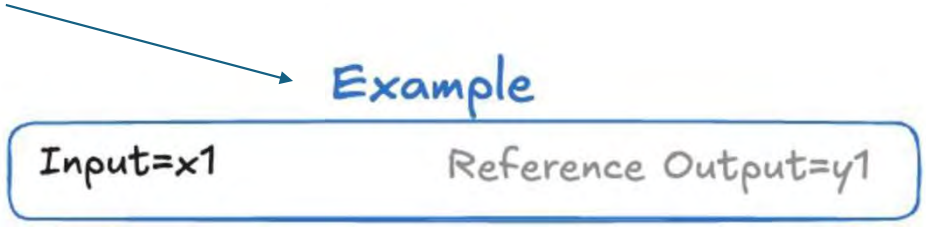
Evaluators



Evaluators

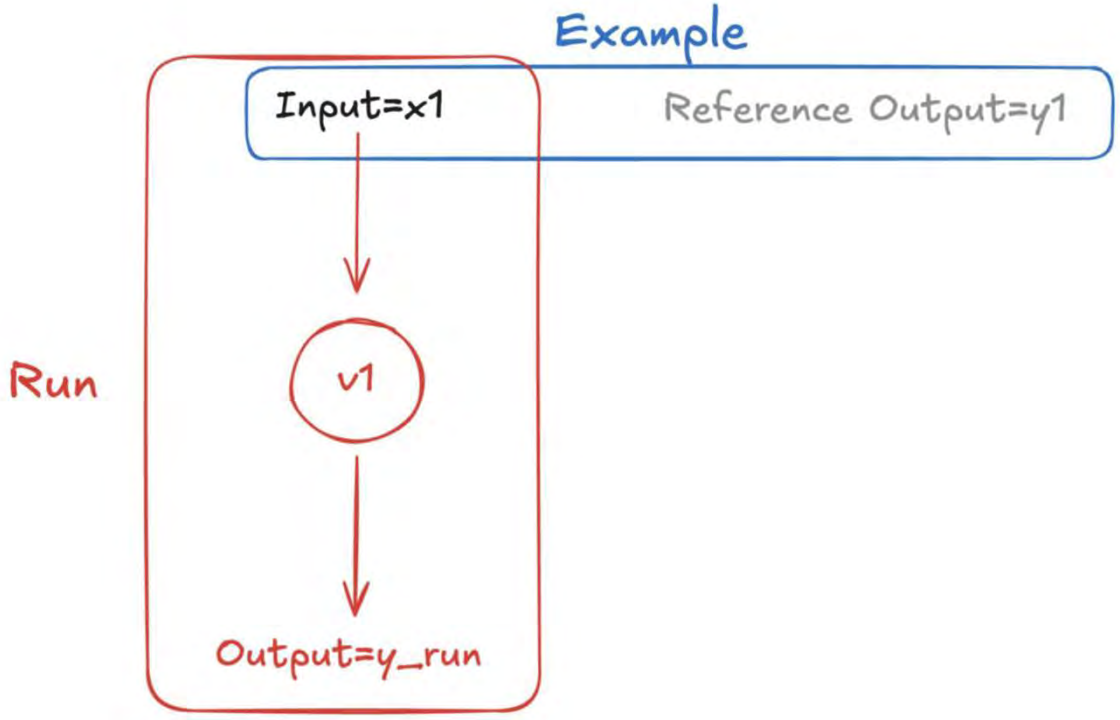
Evaluators compare the dataset **example** against the output **run** from your app

From a Trace
- Manually Added
- AI Generated
- etc.



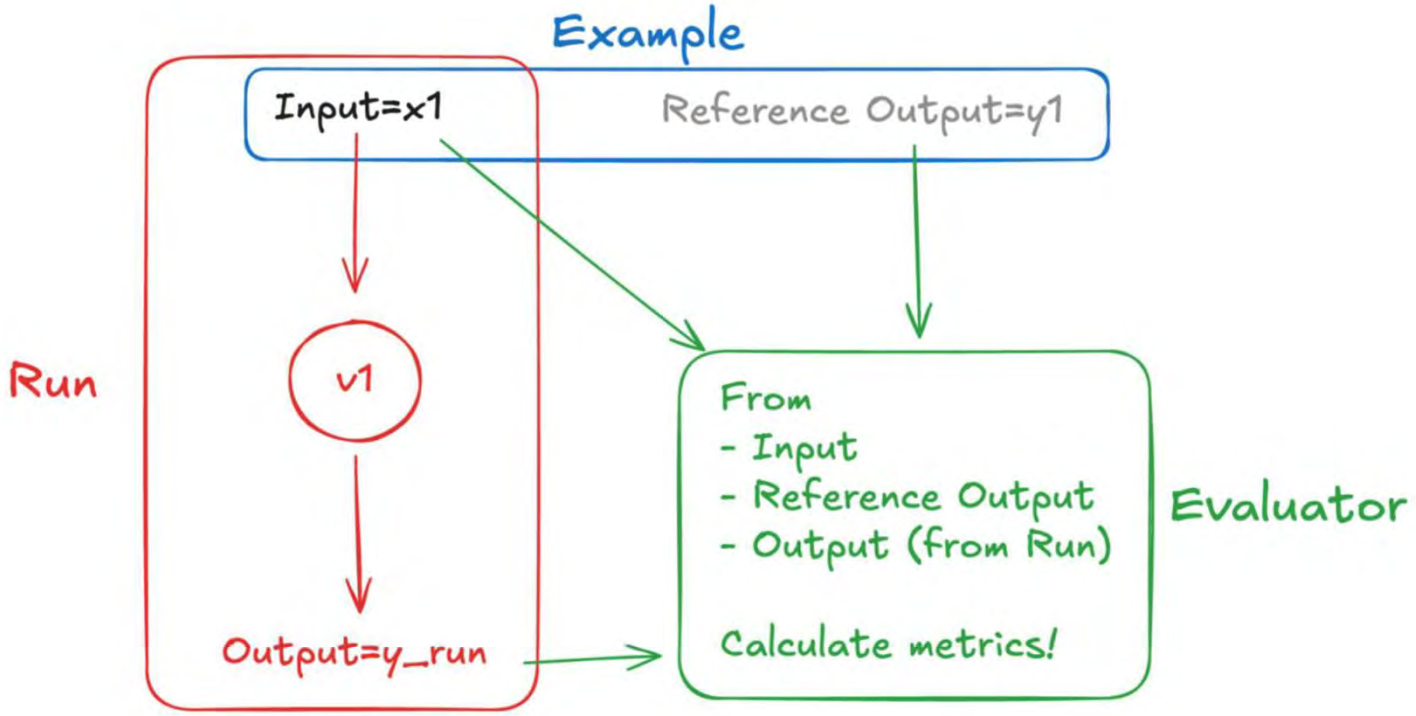
Evaluators

Evaluators compare the dataset **example** against the output **run** from your app



Evaluators

Evaluators compare the dataset **example** against the output **run** from your app



Activity – Evaluators

01 **Set up Environment variables**

02 **LLM-as-Judge Evaluation**

03 **Extend BaseModel from pydantic**

04 **Define evaluator in the UI**

05 **Define Custom Code Evaluator in UI**

Prepare your evaluator for your app in Lesson 13.

Share your dataset url to the padlet

10 mins

Note: Better to write your own custom evaluator like what we showed earlier rather then using the UI to add custom evaluator

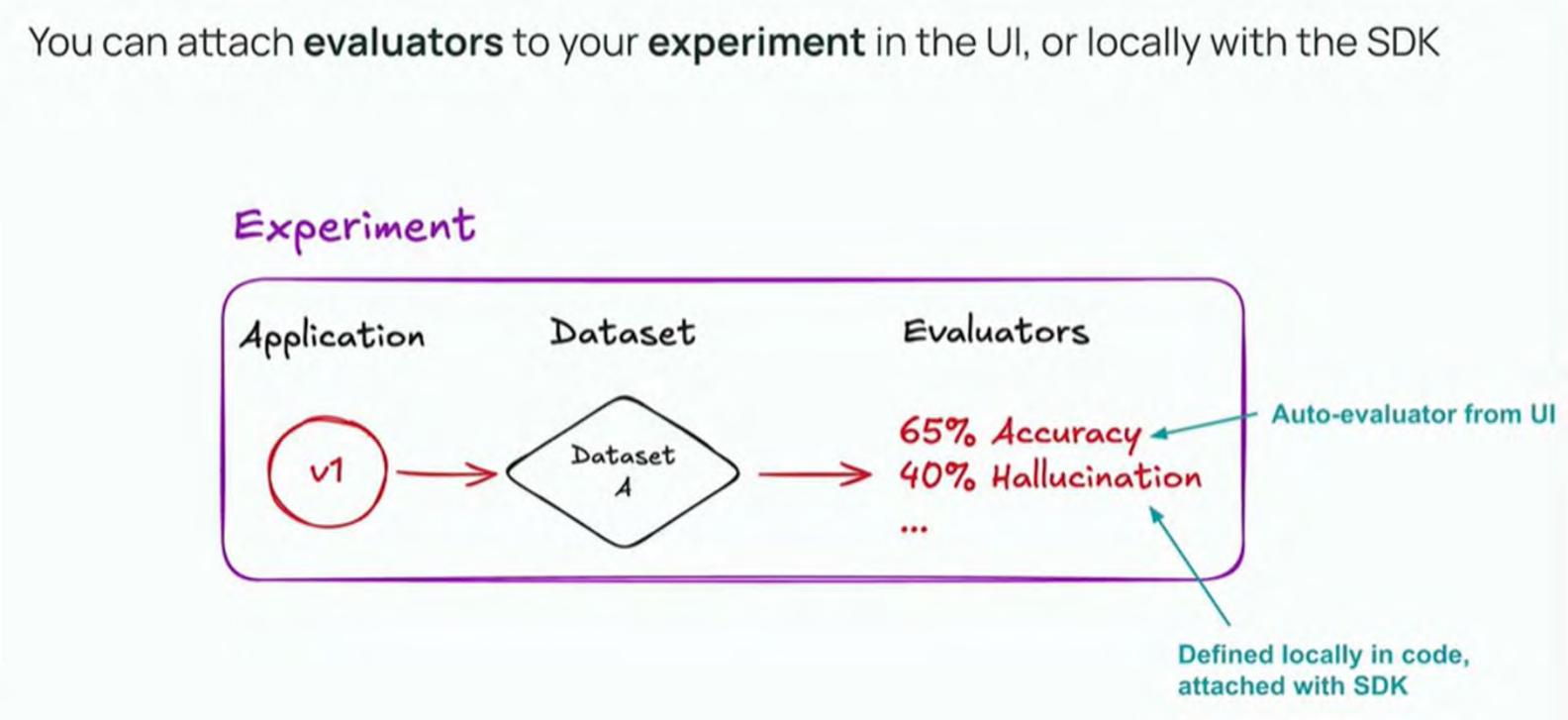
Target to finish by XX:XX

Takeaways

- Evaluators calculate metrics based on a **Run** and an **Example**
  - Specifically, from the Input, Reference output, and Run Output
- You can **define** Evaluators directly in your local code
- You can also define Evaluators in the LangSmith UI
  - **LLM-as-judge evaluators**
  - **Custom code evaluators**

Experiments

Experiment: Running your **application** over a **dataset**, and **evaluating** performance



Activity – Experiments

01 **Set up Environment variables**

02 **Run experiment with gpt-4o and gpt-3.5-turbo**

03 **Run experiment with different data versin**

04 **Run experiment with splits**

05 **Specify Data Points**

06 **Other Parameters**

Prepare and execute your experiment for your app in Lesson 13.

10 mins

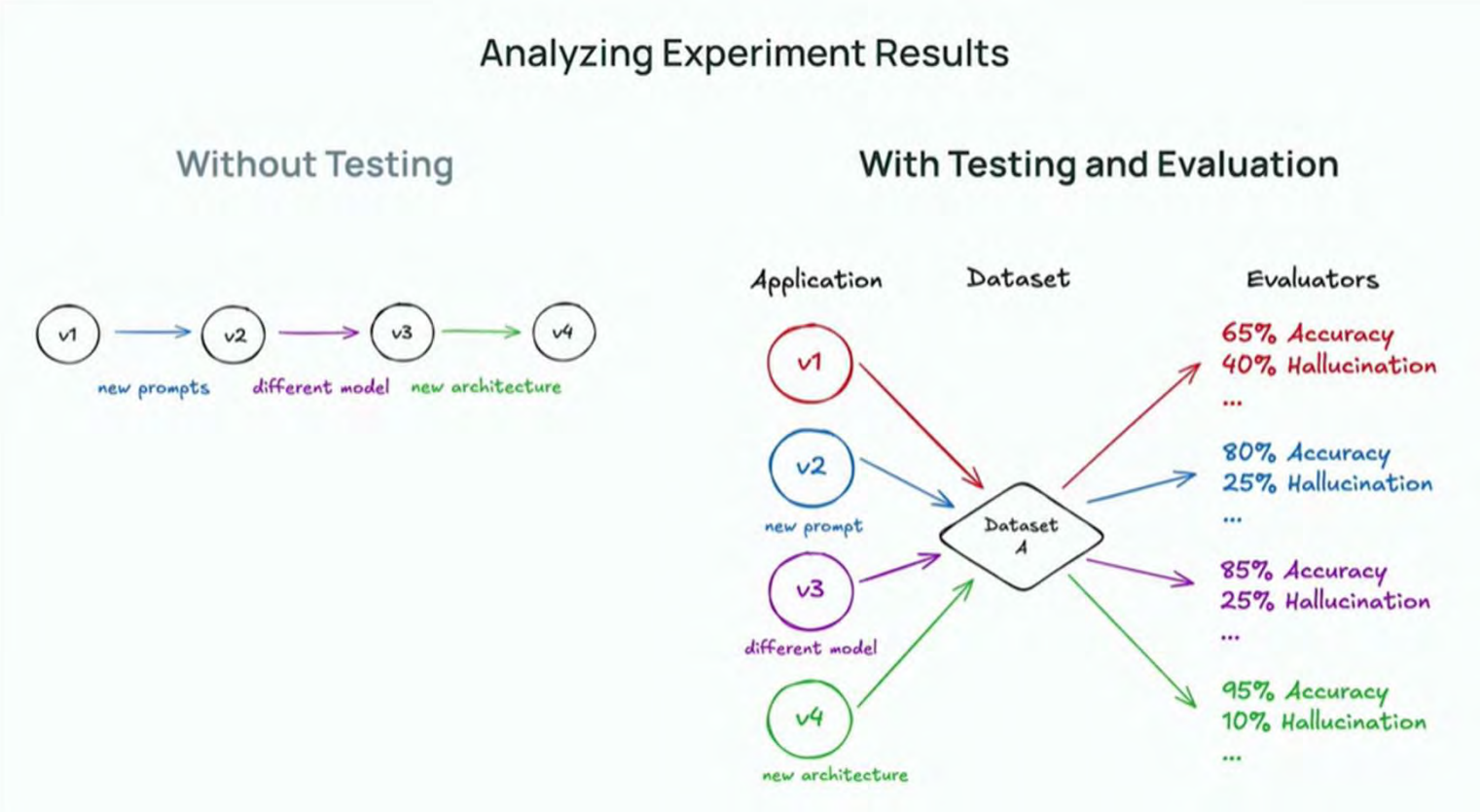
Share your experiment result to the padlet

Target to finish by XX:XX

Takeaway

- **Experiment**: Running your application over a dataset, and evaluating its performance
- Experiments can be run using the LangSmith SDK with **evaluate()**
- Experiments can be run over an entire **dataset** OR
  - A specific version
  - A split
  - Specific examples
- Experiments can be run with **other parameters**
  - Repetitions
  - Concurrent Threads
  - Metadata

Analyzing Experiments results



Activity – Analysing Experiments results

01 Login to Langsmith portal

02 Filter by metadata

03 Go into specific experiment

04 Show Display options

05 View Run/Trace from experiment

10 mins

Target to finish by XX:XX

Takeaway

- Experiments are useful for seeing trends in your application performance as you improve it over time
- You can deep dive into a single experiment and look into how each individual run performed on the dataset example
- You can compare multiple experiments side-by-side and see how they scored on your evaluator metrics
- Experiments give you hard empirical data to push changes to production with confidence

60 mins Lunch Break

Lunch break 11:35 -12:35

Thank you!