11

Evaluating and Testing Generative AI Models

Google Cloud

The information in this presentation is classified:

Google confidential & proprietary

⚠️ This presentation is shared with you under NDA.

- Do **not** record or take screenshots of this presentation.
- Do **not** share or otherwise distribute the information in this presentation with anyone **inside** or **outside** of your organization.

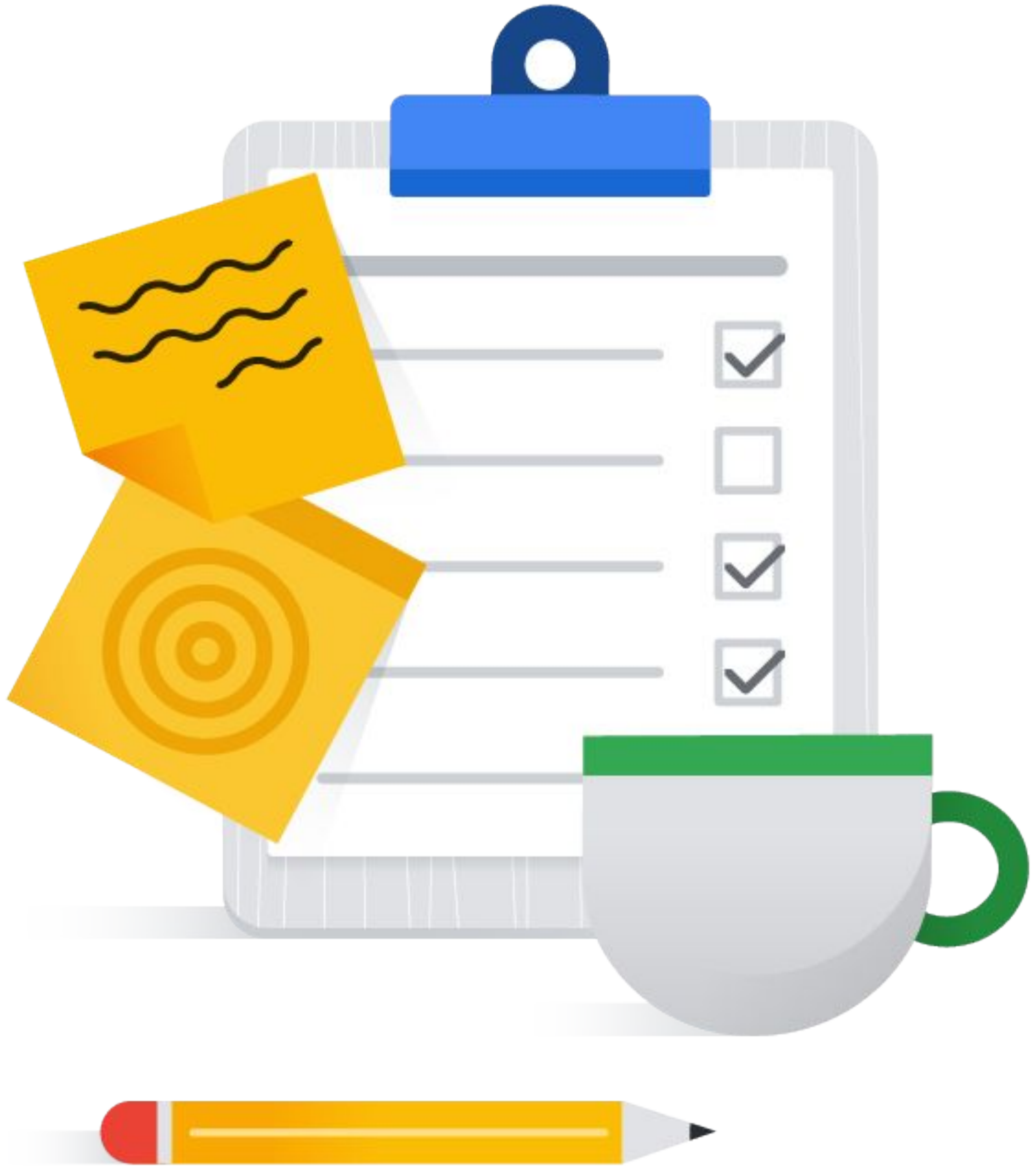
Thank you!

In this module, you learn to ...

01 Evaluate Generative AI applications

02 Write and run automated tests for Generative AI apps

Topics



A State of the Art Challenge: Assessing Generative Model Output Quality

- The quality of output is often subjective:
  - “Write me a clever social media post about the new product line”
- LLMs will return incorrect responses
  - These are called hallucinations
  - It would be great to be able to score the “correctness” of a model
- LLMs do not always return the exact same results
  - This makes it challenging to write automated tests

Which is the best answer?

Summary 1

You can restore a user account (including administrator accounts) up to 20 days after deleting it. After 20 days, the data is gone and you can't restore it.
To restore a user, you must be signed in as a super administrator.
On the Admin console home page, you can see the total number of recently deleted users in the Users section. To go directly to the deleted users page, click that number, and then go to step 4 below.
In the Admin console, go to Menu DirectoryUsers.
Click More options Recently deleted users.
Point to the user, click Recover.
Review the confirmation notice, click Continue.
Select an organizational unit to assign the user to.
Click Recover.
Note: Usually, it takes up to 24 hours for this change to take effect, but in rare cases it can take up to 5 days.

Summary 2

You can restore a user account (including administrator accounts) up to 20 days after deleting it. After 20 days, the data is gone and you can't restore it.
To restore a user, you must be signed in as a super administrator.

1. Sign in to the Admin console.
2. Go to Menu > Directory > Users.
3. Click More options > Recently deleted users.
4. Point to the user and click Recover.
5. Review the confirmation notice and click Continue.
6. Select an organizational unit to assign the user to.
7. Click Recover.

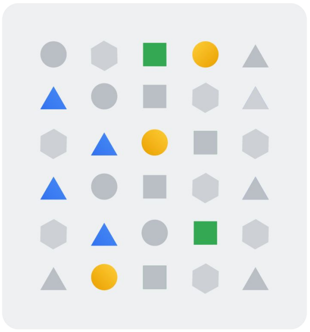
Note: Usually, it takes up to 24 hours for this change to take effect, but in rare cases it can take up to 5 days.

Summary 3

- Restoring user accounts possible only for super administrator and only up to 20 days after deleting it

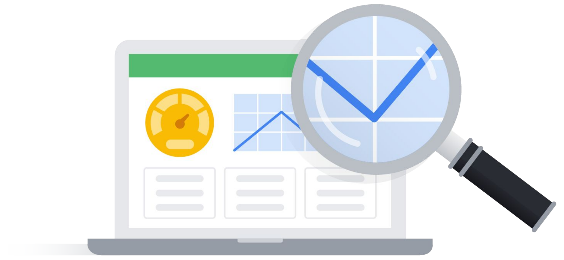
- Process (usually takes 24h, but can take up to 5 business days):
1. Go to Admin console > Menu > Directory > Users
2. Click More options > Recently deleted users
3. Point to the user and click Recover.
4. Review the confirmation notice and click Continue.
5. Click an organizational unit to assign the user.
6. Select Recover.

For Gen AI tasks, evaluation is not a solved problem



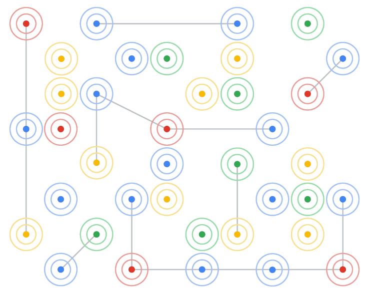
Response quality is hard to measure

It can be hard to automate determining which answer is better and by how much.



Metrics are being updated

Traditional metrics based on exactly matching a known output are becoming dated.



Large decision space

There are many ways to answer many questions posed to generative AI.

The Generative AI evaluation service offers two evaluation paradigms: snapshots or comparisons

Pointwise evaluations

evaluate a snapshot of a single model.

Pairwise evaluations

compares two models to select a preferred one.

And two types of metrics: comparisons to ground truth or generative model-based analysis

Computation-based

compare a model's output to ground truth.

Model-based

use an autorater model to evaluate another model's output.

The evaluation service provides some predefined computation-based metrics

Exact match | BLEU | ROUGE

The `exact_match` metric computes whether a model response matches a reference exactly.

- **Token limit**: None

Evaluation criteria

Not applicable.

Metric input parameters

Input parameter	Description
`response`	The LLM response.
`reference`	The golden LLM response for reference.

or a variety of predefined templates for model-based metrics

	Text use case	Multi-turn chat use case	Other key use cases
Pointwise	• Fluency • Coherence • Groundedness • Safety • Instruction Following • Verbosity • Text Quality	• Multi-turn Chat Quality • Multi-turn Safety	• Summarization Quality • Question Answering Quality
Pairwise	• Fluency • Coherence • Groundedness • Safety • Instruction Following • Verbosity • Text Quality	• Multi-turn Chat Quality • Multi-turn Safety	• Summarization Quality • Question Answering Quality

The service provides structure for you to generate your own metrics with a rating rubric

```
custom_text_quality = PointwiseMetric(
    metric="custom_text_quality",
    metric_prompt_template=PointwiseMetricPromptTemplate(
        criteria={
            "fluency": (
                "Sentences flow smoothly and are easy to read..."
            ),
            "entertaining": (
                "Short, amusing text that incorporates emojis, exclamations and"
                " questions..."
            ),
        },
        rating_rubric={
            "1": "The response performs well on both criteria.",
            "0": "The response is somewhat aligned with both criteria",
            "-1": "The response falls short on both criteria",
        },),)
```

Most evaluations require prompts and responses in the dataset

```
prompts = [
        "Prompt and context 1"
        "Prompt and context 2",
]

responses = [
        "Model response 1",
        "Model response 2"
]


eval_dataset = pd.DataFrame({
    "prompt": prompts,
    "response": responses,
})
```

Computation-based metrics often require an ideal ground truth reference instead of a prompt

```
references = [
        "Ideal correct response 1"
        "Ideal correct response 2",
]

eval_dataset = pd.DataFrame({
    "response": responses,
    “reference”: references
})
```

Pairwise metrics require a “baseline” other model’s response to compare a new model’s response to

```
baseline_model_responses = [
        "Other model’s response 1"
        "Other model’s response 2",
]


eval_dataset = pd.DataFrame({
    "response": responses,
    “reference”: references,
     "baseline_model_response": baseline_model_responses,
})
```

Results come with Summary Metrics and an example-by-example report with rating explanations

```
In [9]: display_eval_report((("Eval Result", result.summary_metrics, result.metrics_table)))
```

Eval Result

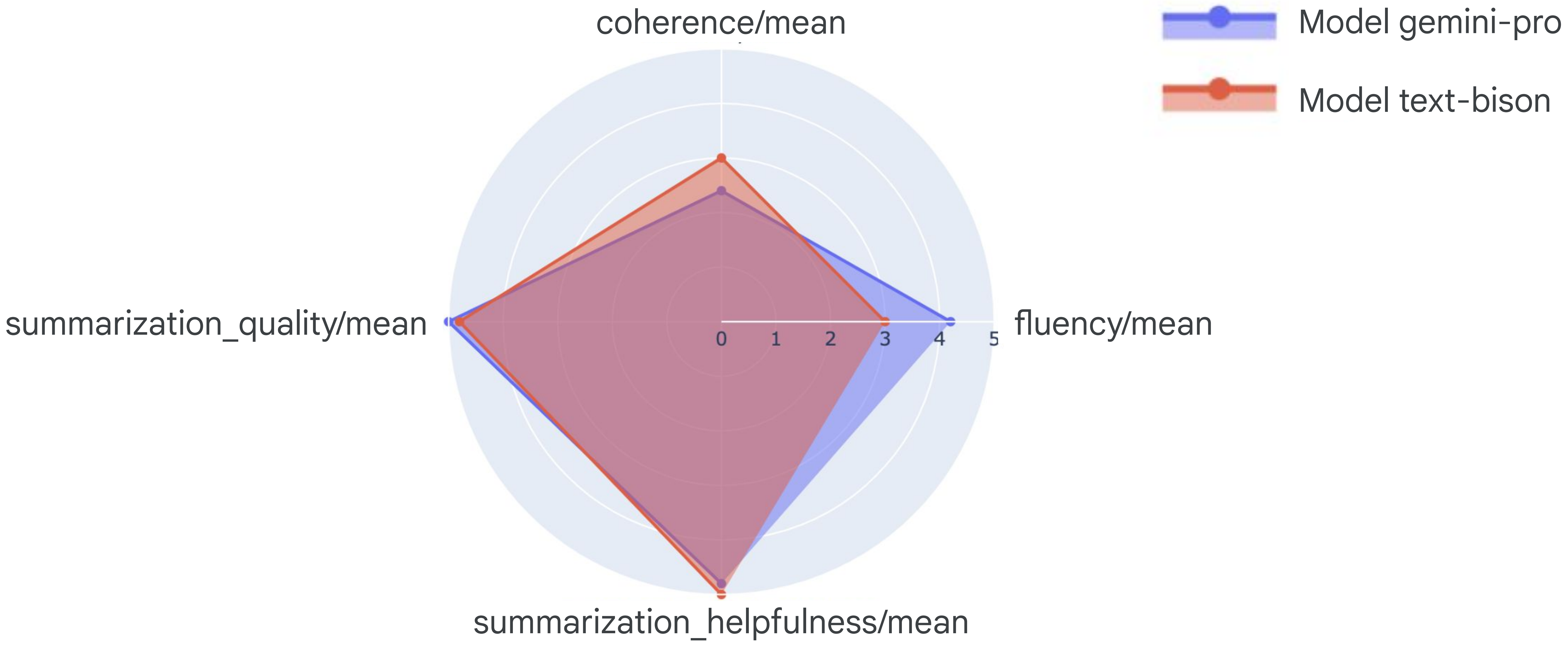
Summary Metrics

	row_count	safety/mean	safety/std	coherence/mean	coherence/std	fluency/mean	fluency/std
0	7.0	1.0	0.0	2.285714	0.755929	4.428571	0.534522

Report Metrics

content	response	safety/explanation	safety/confidence	safety	coherence/explanation
What commonly inspires individuals to pursue t...	**Intrinsic Factors:**\n\n* **Passion and Inte...	The response does not contain any hate speech,...	1.0	1.0	The response lacks a clear structure and logic...
In general, how do professionals approach prob...	**Professionals approach problem-solving in th...	The response does not contain any hate speech,...	1.0	1.0	The response provides a clear and logical prog...
Can you provide an example of a significant ch...	**Significant Challenge:** Maintaining Work-Li...	The response does not contain any hate speech,...	1.0	1.0	The response lacks a clear logical flow. While...

You can plot models' performance to compare models



Some computation-based metrics to know: ROUGE is used for summarization and translation

Compare an automatically produced summary or translation against a reference (human-produced) summary or translation

- ROUGE-N: Overlap of n-grams between the system and reference summaries
- ROUGE-L: Longest Common Subsequence (LCS) based statistics
- ROUGE-S: Skip-bigram based co-occurrence statistics
- ROUGE-W: Weighted LCS-based statistics that favors LCSs

ROUGE-L: Longest Common Subsequence

- A longest common subsequence (LCS) is the longest subsequence common to all sequences in a set of sequences (often just two sequences)
- Consider the sequences (ABCD) and (ACBAD), they have:
  - 5 length-2 common subsequences: (AB), (AC), (AD), (BD), and (CD)
  - 2 length-3 common subsequences: (ABD) and (ACD)
  - So (ABD) and (ACD) are their longest common subsequences
- ROUGE_L returns a value between 0 and 1
  - 1 means the sequences are the same
  - 0 means the sequences have nothing in common
  - Closer to 1 is better

BLEU (Bilingual Evaluation Understudy) is a metric used to evaluate the quality of machine-generated text

- Compares generated text to reference human-generated text
  - Used for tasks such as machine translation, text summarization, and image captioning
- Ranges from 0 to 1
  - 0 indicates no overlap between the machine-generated text and the reference text
  - 1 indicates a perfect match

BLEU - Interpretation

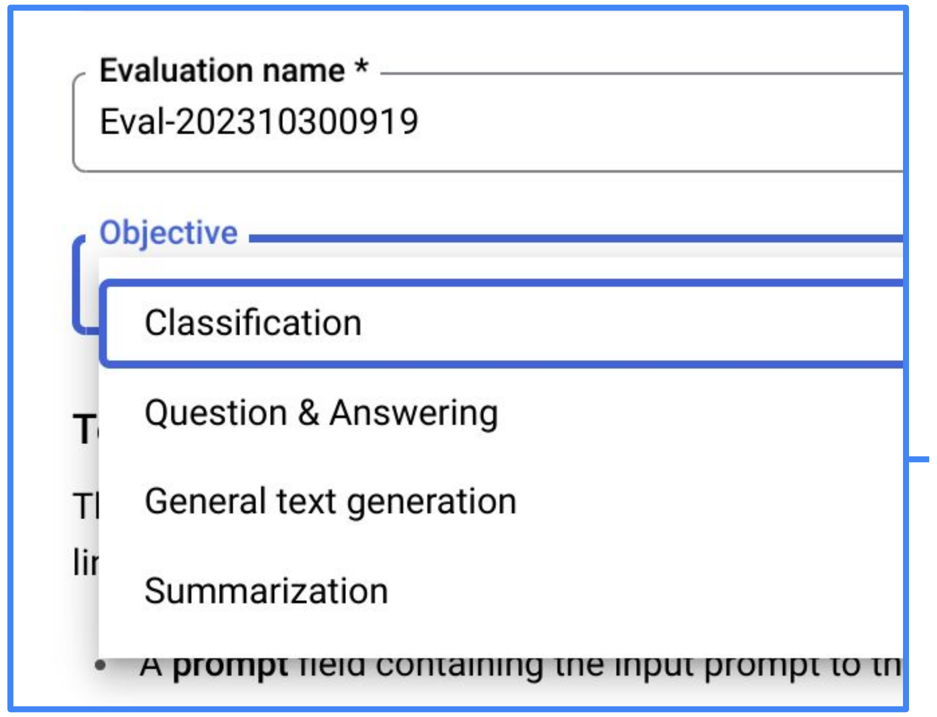
BLEU Score [%]	Interpretation
< 10	Almost useless
10 - 19	Hard to get the gist
20 - 29	The gist is clear, but has grammatical errors
30 - 39	Understandable to good
40 - 49	High quality
50 - 60	Very high quality
> 60	Quality often better than human

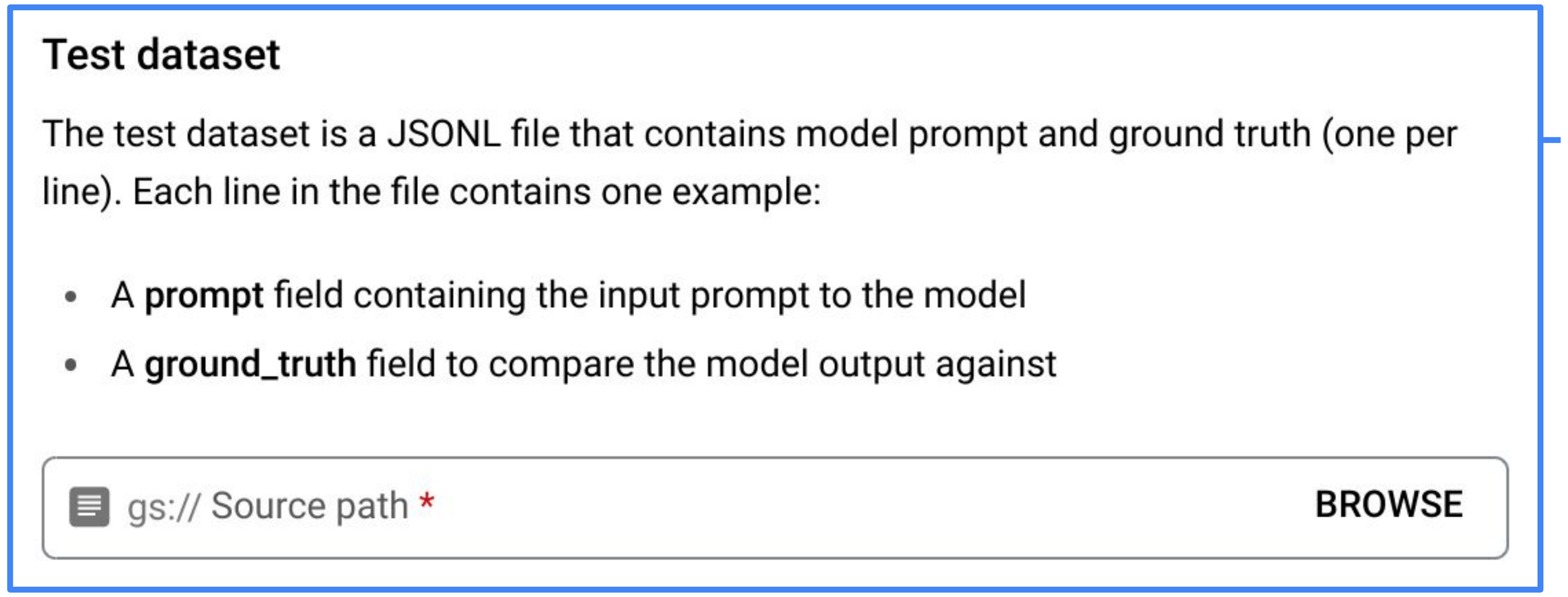
BLEU scores from different corpora and languages **cannot** be directly compared.

Exact Match measures the percentage of predictions that match any one of the ground truth answers exactly

- For each question and answer pair, if the characters of the model's prediction exactly match the characters of (one of) the True Answer(s), then EM = 1, otherwise EM = 0
- This is a strict all-or-nothing metric
  - Being off by a single character results in a score of 0
- This metric is limited in that it outputs the same score for something that is completely wrong as for something that is correct except for a single character
- These traditional NLP metrics looking for exact matches are good for short completions or short phrases of Question-Answering results, but are harder to rely on for longer answers where there can be many ways to express something well

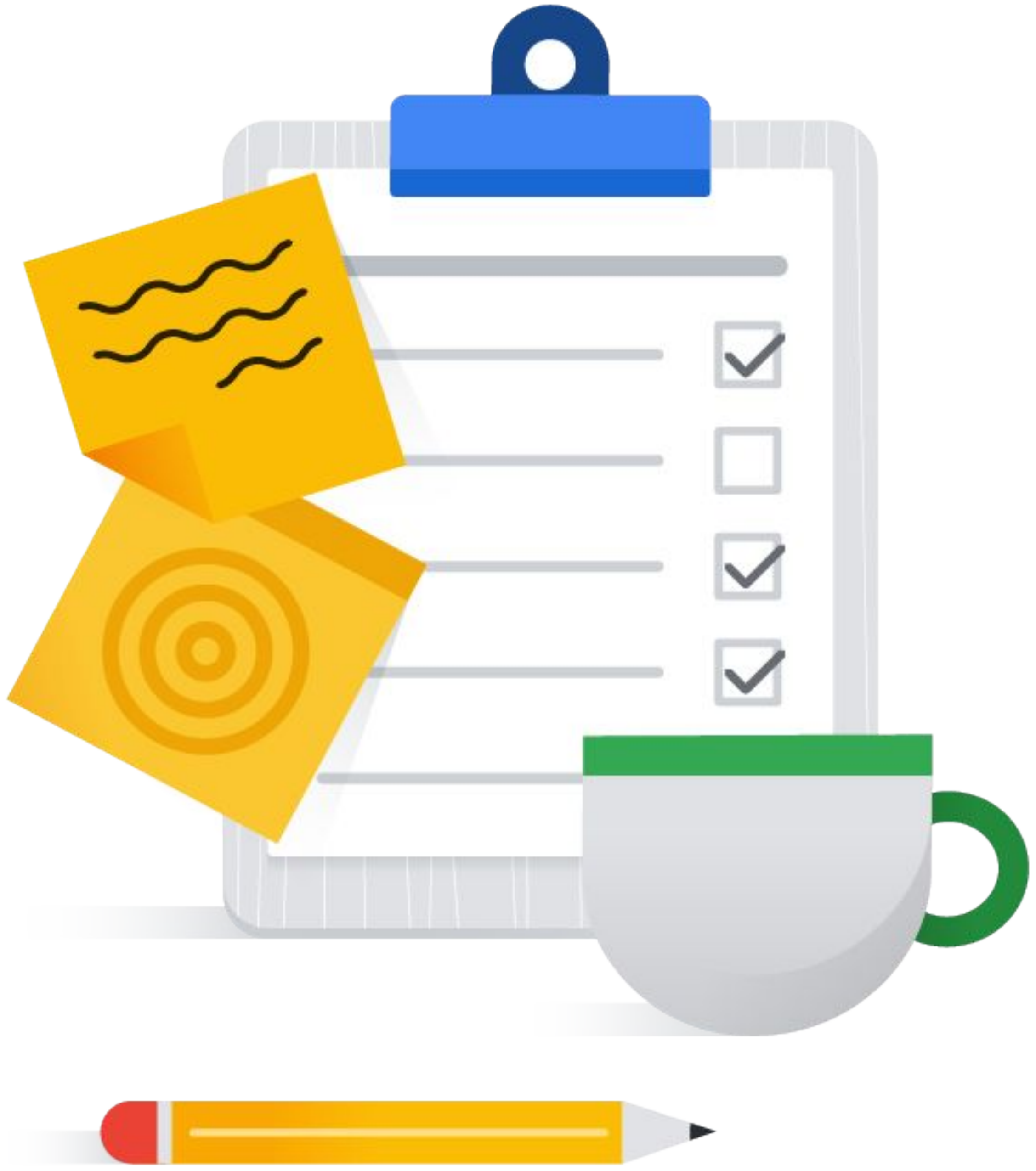
You can also evaluate specific tasks for a trained or fine-tuned model using Vertex AI Model Registry





1. Select the model to view its Details and click Create Evaluation
2. Choose the model objective
3. Set the location of the test dataset
   - JSON-L file with `prompt` and `ground_truth` fields

Topics



You should unit test your LLM applications

- Unit testing is straightforward with classification examples where there is only one well-defined answer
  - Compare the actual result to the expected result
  - Thorough unit testing will quickly expose where prompt-tuning, more examples, and model fine-tuning might be needed
- Unit testing is more difficult in cases where there isn't a single right answer
  - One strategy is to use the model to determine if two answers are fundamentally equivalent without being exactly the same

Testing for an Expected Response

```
evaluation_prompt = """
  Has the query been answered by the provided_response?
  The new tractor model is the Arcturus.
  Respond with only one word: yes or no

  query: {query}
  provided_response: {provided_response}
  evaluation: """
```

Testing for a Fallback Response

```
evaluation_prompt = """
  Does the response decline to discuss a non-farming related topic
  and encourage the user to ask about farming instead?
  Respond with only one word: yes or no

  query: {query}
  provided_response: {provided_response}
  evaluation: """
```

Testing for groundedness

```
evaluation_prompt = """
  Does the provided_response answer the query
  as well as possible without adding information
  that does not appear in the context?
  Respond with only one word: yes or no

  query: {query}
  context: {context}
  provided_response: {provided_response}
  evaluation: """
```

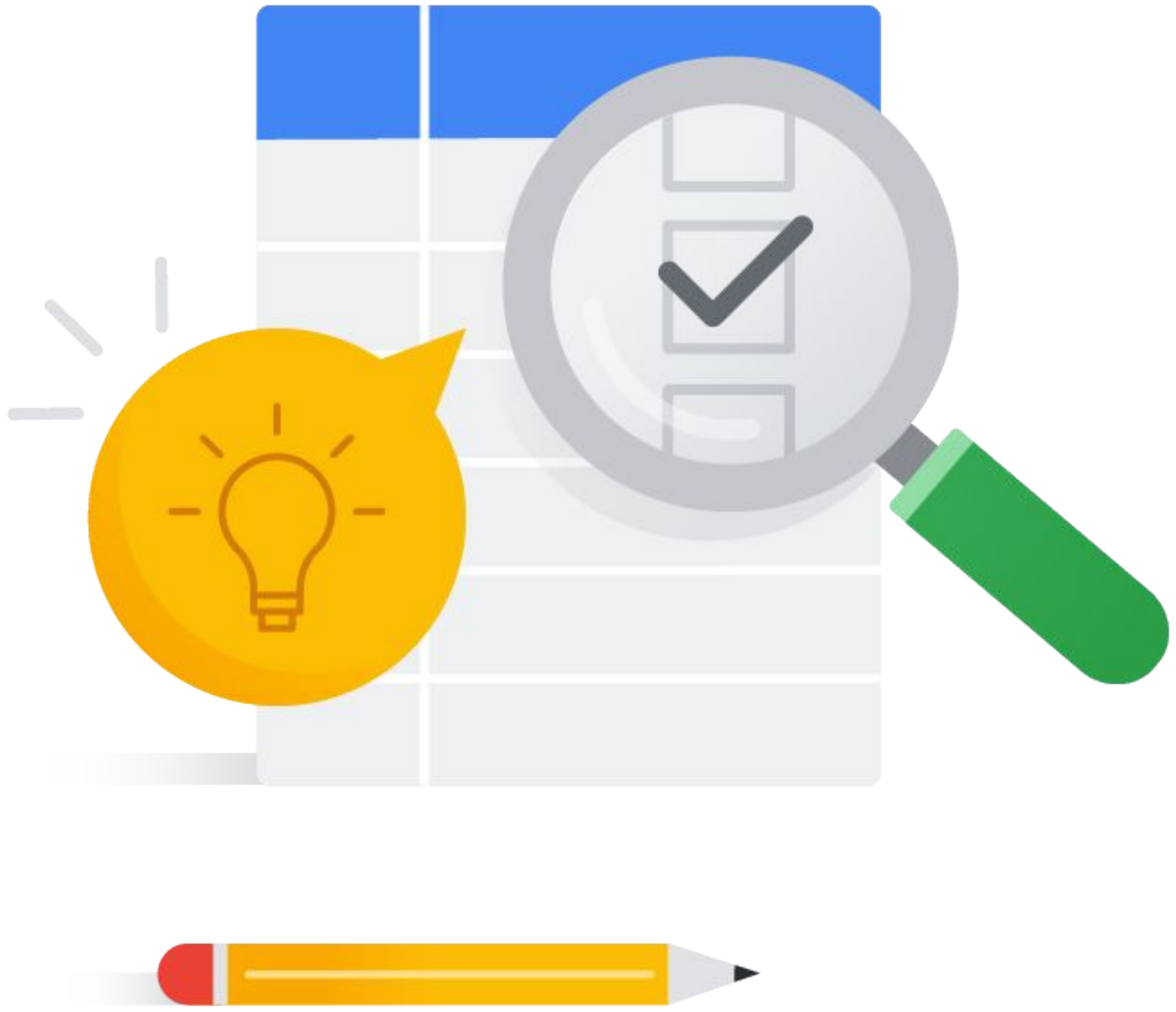
Testing for matching

```
evaluation_prompt = """
  Compare the following Tweets. Are they fundamentally the same?

   Only return Yes or No

   Tweet 1: {0}
   Tweet 2: {1}
   Output:

 """
```

Lab

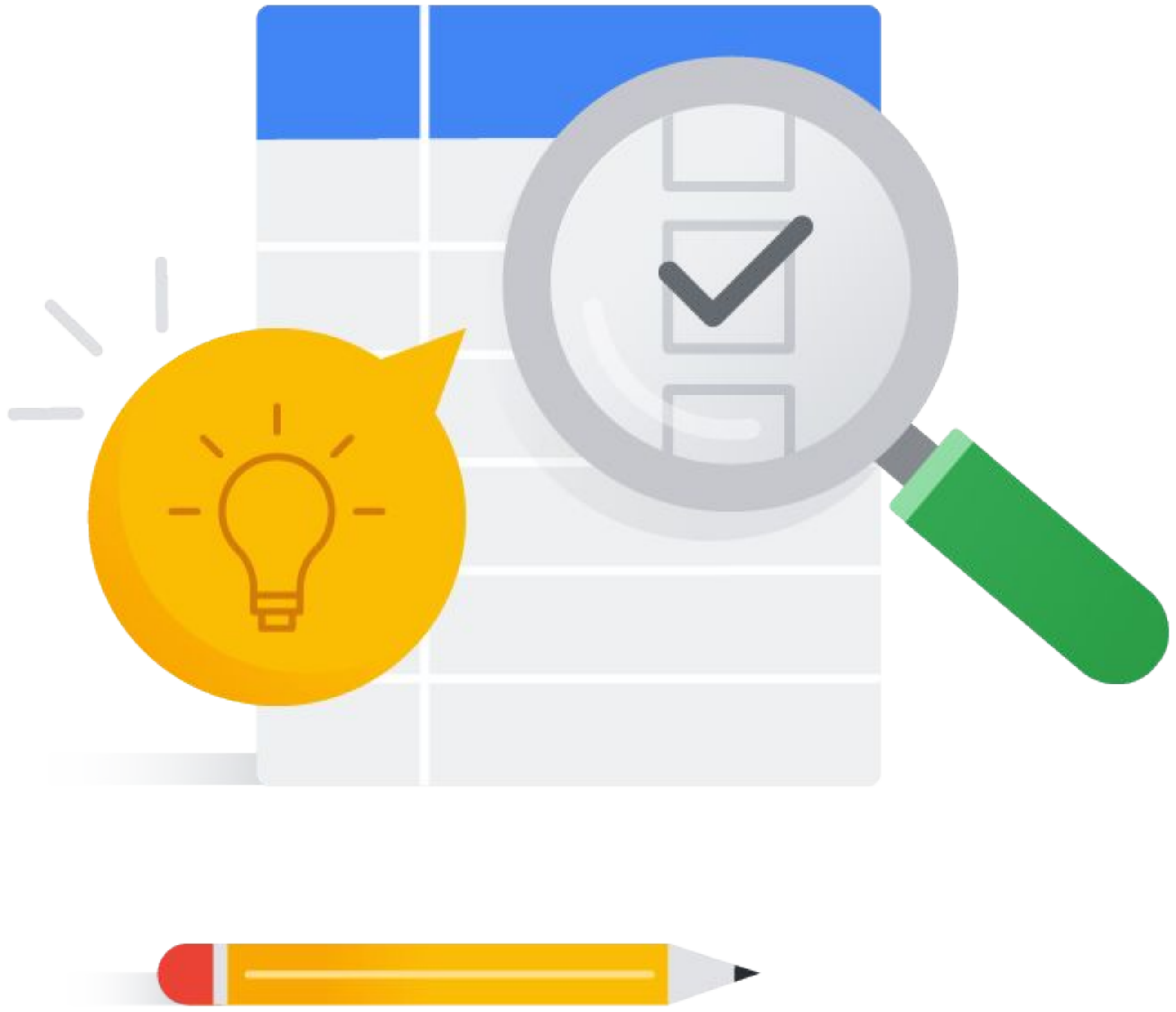
30 min

Lab: Evaluating ROUGE-L Text Similarity Metric



Lab

1 hour



Lab: Unit testing generative AI applications

In this module, you learned to ...

01 Evaluate Generative AI applications

02 Write and run automated tests for Generative AI apps

Questions and answers

Quiz question

What might be a good evaluation metric for a Classification problem?

A: RMSE

B: F1

C: ROUGE-L

D: BLEU score

Quiz question

What might be a good evaluation metric for a Classification problem?

A: RMSE

B: F1

C: ROUGE-L

D: BLEU score

Quiz question

What might be a good evaluation metric for a text generation problem? (Choose two)

A: RMSE

B: F1

C: ROUGE-L

D: BLEU score

Quiz question

What might be a good evaluation metric for a text generation problem? (Choose two)

A: RMSE

B: F1

C: ROUGE-L

D: BLEU score

Quiz question

When using F1, ROUGE-L, or BLEU to evaluate model versions, how do you know which is better?

A: Closest to 0 is best

B: Closest to 1 is best

C: The greater the number the better

D: The smaller the number the better

Quiz question

When using F1, ROUGE-L, or BLEU to evaluate model versions, how do you know which is better?

A: Closest to 0 is best

B: Closest to 1 is best

C: The greater the number the better

D: The smaller the number the better

Google Cloud