

Testing a Fine-Tuned Radiology Model

Text similarity and evaluation metrics | Practical guide for chest X-ray + Findings to Impression generation, including Google Colab code.

1. Testing task

For each held-out example, provide the chest X-ray image and Findings to the model. Compare the generated Impression with the radiologist-written reference Impression. Keep the reference Impression out of the model input.

Example: Reference: "No acute cardiopulmonary abnormality." Generated: "No evidence of acute heart or lung disease." These differ in wording but have similar meaning. By contrast, "no pleural effusion" versus "small pleural effusion" is a clinically important error.

2. Recommended metrics

Metric	Measures	Use
ROUGE-L	Longest shared word sequence	Lexical overlap
BLEU	Matching word n-grams with brevity penalty	Wording similarity
BERTScore F1	Contextual embedding similarity	Semantic similarity
Clinical review / RadGraph / CheXbert	Abnormalities, negation, laterality and relationships	Clinical correctness

Recommendation: Start with ROUGE-L and BERTScore F1, optionally include BLEU, and add clinical correctness checks. Text similarity is not proof of medical accuracy.

3. Prepare a held-out test dataset

Start with 100-200 examples for pipeline testing, then evaluate on a larger representative set. Exclude test examples from training and model selection. Group all images from one study together; use patient-level separation when possible.

```
{"id": "001", "image": "images/xray001.png",  
  "findings": "The lungs are clear. No pleural effusion.",  
  "impression": "No acute cardiopulmonary abnormality."}
```

This JSONL record is illustrative, not a real ReXGradient record. The evaluation script may use a different file layout than the model API.

4. Generate predictions

For every test example, submit the image and Findings to the fine-tuned model, save the generated Impression, and retain the reference Impression separately. Use identical prompt and inference settings for the base and fine-tuned models. Export **predictions.csv** with columns **id**, **reference**, **prediction**.

5. Calculate metrics in Google Colab

Step A - Install packages

```
!pip install -q evaluate rouge_score bert_score sacrebleu pandas
```

Step B - Upload predictions.csv

```
from google.colab import files
import pandas as pd
files.upload()
df = pd.read_csv("predictions.csv")
df = df.dropna(subset=["reference", "prediction"]).copy()
print("Test examples:", len(df))
```

Step C - Compute corpus-level metrics

```
import evaluate
rouge = evaluate.load("rouge")
bleu = evaluate.load("bleu")
bertscore = evaluate.load("bertscore")
refs = df["reference"].astype(str).tolist()
preds = df["prediction"].astype(str).tolist()
r = rouge.compute(predictions=preds, references=refs)
b = bleu.compute(predictions=preds,
                  references=[[x] for x in refs])
bs = bertscore.compute(predictions=preds,
                       references=refs, lang="en")
print("ROUGE-L:", round(r["rougeL"], 4))
print("BLEU:", round(b["bleu"], 4))
print("BERTScore F1:", round(sum(bs["f1"])/len(bs["f1"]), 4))
```

Step D - Save individual example scores

```
df["BERTScore_F1"] = bs["f1"]
df["ROUGE_L"] = [
    rouge.compute(predictions=[p], references=[r])["rougeL"]
    for p, r in zip(preds, refs)
]
df.to_csv("evaluation_results.csv", index=False)
print(df[["id", "ROUGE_L", "BERTScore_F1"]].head())
```

BLEU is best reported at corpus level; individual short Impression BLEU scores can be unstable.

6. Interpret the scores

Metric	Hypothetical score	Interpretation
ROUGE-L	0.71	Substantial lexical overlap
BLEU	0.46	Some exact n-gram agreement
BERTScore F1	0.91	High contextual similarity

These values are illustrative, not actual evaluation results. There are no universal clinical-correctness thresholds for these metrics. Compare models on the same examples with the same implementation and settings.

7. Check clinical correctness

Reference	Generated	Potential error
No pneumothorax	Small pneumothorax	Negation / false finding
Right pleural effusion	Left pleural effusion	Laterality

Reference	Generated	Potential error
Mild cardiomegaly	Severe cardiomegaly	Severity
No focal consolidation	No acute abnormality	Potentially acceptable; inspect context

Consider RadGraph F1 for clinical entities and relationships, CheXbert for chest X-ray observation labels, and radiologist review for clinically significant errors. An automated semantic score cannot substitute for clinical review.

8. Compare base and fine-tuned models

Metric	Base (example)	Fine-tuned (example)
ROUGE-L	0.58	0.71
BLEU	0.31	0.46
BERTScore F1	0.85	0.91
Clinically significant error rate	12%	7%

All numbers above are hypothetical. Define the clinical error rubric before measuring error rates. Use the same held-out dataset for both models.

9. Testing report checklist

Document the dataset and split, number of examples, model versions and inference settings, ROUGE-L/BLEU/BERTScore F1, at least five example predictions including errors, clinical correctness review, and a conclusion about improvement over the base model.

Special note: The same broad workflow applies to image question-answering, but the metrics depend on the output. Short constrained answers may benefit from exact-match accuracy; open-ended answers may require semantic metrics and task-specific correctness checks.

References and resources

ReXGradient-160K dataset: <https://huggingface.co/datasets/rajpurkarlab/ReXGradient-160K>

Hugging Face Evaluate: <https://huggingface.co/docs/evaluate/index>

RadGraph: <https://physionet.org/content/radgraph/> ; BERTScore: https://github.com/Tiiiger/bert_score