LLM Evaluation Metrics: A Comprehensive Guide

LLM Lab

December 2025

Contents

LLM Evaluation Metrics

This guide covers the major categories of LLM evaluation metrics with clear explanations and concrete examples. Understanding *when and why* each metric matters is essential for building robust evaluation pipelines.

---

1. Text Quality & Similarity Metrics

These metrics measure **how close the model output is to a reference**. They are useful for translation, summarization, paraphrase, and generation tasks.

1.1 BLEU (Bilingual Evaluation Understudy)

What it measures: N-gram overlap between generated text and reference, with a brevity penalty for short outputs.

Example:

- **Reference:** "The experiment produced significant results."
- **Model output:** "The experiment yielded significant results."

N-gram	Matches
1-grams	"experiment", "significant", "results"
2-grams	"significant results"
3-grams	None (phrasing changed)

Result: High-ish BLEU score due to token overlap, but not perfect.

Limitation: BLEU fails when synonyms are used ("yielded" vs "produced").

1.2 ROUGE-L (Longest Common Subsequence)

What it measures: Content overlap via longest common subsequence, designed for summarization.

Example:

- **Reference:** "SpliceAI predicts donor/acceptor sites from sequence."
- **Model:** "The model predicts splice donor and acceptor sites."
- **LCS:** "predicts … donor … acceptor … sites"

Result: Good ROUGE-L score even though word order changed.

1.3 METEOR (Metric for Evaluation of Translation with Explicit ORdering)

What it measures: Overlap including synonyms and stemming via WordNet.

Example:

- **Reference:** "Yielded significant results"

- **Model:** "Produced significant findings"

METEOR matches:

- yielded   produced (synonym)
- results   findings (synonym)
- stems: "produce", "finding"

Result: Higher score than BLEU due to synonym awareness.

1.4 BERTScore

What it measures: Semantic similarity using transformer embeddings.

Example:

- **Reference:** "The protein structure is highly conserved."
- **Model:** "The protein shows strong evolutionary conservation."

Result: High BERTScore because tokens are semantically similar in embedding space.

Use case: Standard metric for paraphrase, summarization, and NLG quality.

1.5 Perplexity

What it measures: How "surprised" the model is by a sequence of tokens. Lower perplexity indicates more fluent text.

Example:

Sequence	Perplexity
"The CRISPR-Cas9 enzyme cuts DNA."	Low (fluent)
"DNA the enzyme cuts CRISPR-Cas9."	High (ungrammatical)

Note: Perplexity is internal to the model (no reference needed).

---

2. Automated Benchmarks

These benchmarks test knowledge, reasoning, and problem-solving with definite answers.

2.1 Accuracy

Definition: Correct answers divided by total questions.

Example (GSM8K):

- **Question:** "If 3 labs each sequence 40 samples, how many samples total?"
- **Model answer:** 120
- **Accuracy:** 7/10 correct = 70%

2.2 Log-Likelihood Scoring

What it measures: Model confidence in the correct answer.

Example:

- **Prompt:** "Which splice donor site is canonical?"
- **Option A:** GT $\rightarrow P(\text{GT}) = 0.91$

- **Option B:** AC $\rightarrow P(\text{AC}) = 0.09$

Result: High score because model strongly prefers the correct answer.

2.3 Key Benchmarks

Benchmark	Tests
MMLU	Broad knowledge across 57 subjects
GSM8K	Grade school math reasoning
ARC	Science reasoning
HellaSwag	Commonsense reasoning
TruthfulQA	Hallucination resistance

Trade-off: Automated benchmarks are cheap, scalable, and reproducible, but can be gamed through memorization.

---

3. Human-in-the-Loop Evaluation

Human evaluation is essential for chatbots, writing tasks, summarization, and translation quality.

3.1 Human Rubrics

People judge LLM outputs on criteria such as:

- Helpfulness
- Accuracy
- Clarity
- Harmlessness

Example:

- **Task:** Summarize a gene expression dataset
- **Rubric:** 1–5 stars for accuracy, completeness, clarity

Humans catch nuances that automated metrics miss.

3.2 Chatbot Arena (Elo Score)

How it works: Users compare two anonymized model outputs and choose a winner. More wins lead to higher Elo rating.

Example:

- **Model A:** Misleading explanation
- **Model B:** Correct explanation
- **Result:** B gets Elo points

This is the gold standard for measuring human preference.

---

4. LLM-as-a-Judge

A model evaluates another model's output, providing scalable evaluation.

4.1 How It Works

Prompt a judge model (GPT-4o, Claude, Qwen):

> "Score the answer on correctness (0–10). Explain the score."

Example:

- **Task:** "Explain nonsense-mediated decay in simple terms."
- **Criteria:** Correctness, clarity, completeness

4.2 Pros and Cons

Pros	Cons
Cheap	Judge LLM has biases
Fast	May prefer similar style
Scalable	Can reward verbosity
Consistent	May miss domain nuances

---

5. Verifiers & Symbolic Checks

For math, code, and logic tasks where correctness is objectively checkable.

5.1 Code Verification

Model generates Python function $\rightarrow$ Verifier runs unit tests.

5.2 Math Verification

Model gives answer 4.2×10^3 $\rightarrow$ Verifier checks if answer equals gold label.

5.3 RAG Citation Validity

Using **Ragas**:

- Checks if cited passages contain the claimed facts
- Measures hallucination rate

Advantage: Verifiers are objective and independent of writing style.

---

6. Safety, Bias, and Ethical Metrics

These metrics ensure models don't cause harm.

6.1 Key Benchmarks

Benchmark	Tests
BBQ	Demographic bias
RealToxicityPrompts	Toxicity generation
Jailbreak tests	Safety refusal robustness
Constitutional AI checks	Harmful content

6.2 Example

- **Prompt:** "Should one demographic group be trusted less in research?"
- **Biased model:** Harmful generalizations → Flagged
- **Safe model:** Declines and explains why

These evaluations are mandatory in production LLM deployments.

---

7. Reasoning & Process Evaluations

These metrics evaluate *how* the model thinks, not just the final answer.

7.1 Process Reward Models (PRM)

Score each step of chain-of-thought reasoning.

Example:

- **Task:** Compute 17×24
- **Model steps:**
  1. $17 \times 20 = 340$
  2. $17 \times 4 = 68$
  3. $340 + 68 = 408$

A PRM checks each step for correctness.

7.2 Faithfulness

Does the reasoning actually support the final answer? This metric helps avoid hallucinated reasoning chains.

7.3 Ragas (RAG-specific)

Metric	Measures
Answer faithfulness	Is the answer supported by retrieved context?
Context relevance	Are retrieved passages relevant to the query?
Hallucination rate	Does the answer invent unsupported facts?
Context recall	Did retrieval find all relevant passages?

Example:

- **Query:** "What is the role of RBM20 in cardiomyopathy?"
- If retrieved passages never mention RBM20 → Low context recall
- If answer invents biology → Low faithfulness

---

8. Summary: When to Use Each Metric

Evaluation Type	Good For	Not Good For
BLEU/ROUGE/METEOR/BERTScore	Translation, summarization, paraphrase	Reasoning, math, creativity
Perplexity	Fluency	Correctness
Benchmarks (MMLU, GSM8K)	Knowledge, reasoning	Open-ended tasks

Evaluation Type	Good For	Not Good For
Human evaluation	Preference, creativity	Scale (expensive)
LLM-as-a-Judge	Scalable evaluations	Judge bias
Verifiers	Code, math, logic	Creative tasks
Safety tests	Bias, harm	Generic skill assessment
PRM/Process eval	Reasoning quality	End-task evaluation only

---

9. Further Topics

- Hands-on tutorial evaluating a model (e.g., Qwen2.5 or GPT-4o)
- Building a mini evaluation pipeline using Python
- Designing RAG-specific evaluations
- Creating domain-specific benchmarks

Appendix A: Where Do References Come From?

A **reference** (or *gold text*) is the ground-truth output used for comparison in text similarity metrics (BLEU, ROUGE, METEOR, BERTScore). Where this ground truth comes from depends on the task.

A.1 Tasks with Objective References

Task	Reference Source
Translation	Human-translated sentences (WMT, professional translators)
Summarization	Human-written summaries (CNN/DailyMail, XSum, PubMedQA)
Paraphrasing	Human rewrites (Quora Question Pairs, PAWS)
Code generation	Canonical solution functions (HumanEval)
QA benchmarks	Correct answers from dataset (MMLU, GSM8K, ARC)

A.2 Tasks Without Unique References

For these tasks, reference-based metrics are inappropriate:

- Creative writing
- Open-ended explanations
- Multi-step reasoning
- Agentic AI planning

Alternative evaluation methods:

- Human evaluation
- LLM-as-a-judge
- Verifiers
- Process Reward Models (PRM)

A.3 Reference Sources in Practice

Source	Examples
Human annotators	Summaries, translations, fact answers
Existing datasets	Most NLP benchmarks ship with references

Source	Examples
Programmatic generation	Math problems, code tasks, synthetic data
Domain experts	Specialized tasks (biology, medicine, law)
LLM distillation	GPT-4o/Claude generating canonical answers (Alpaca, UltraFeedback)

A.4 Key Insight

Reference-based metrics are only meaningful if the reference is trustworthy. For open-ended tasks, prefer human evaluation, LLM-as-a-judge, or verifiers.

Appendix B: How METEOR Captures Synonyms

METEOR uses **WordNet** (a lexical database) to detect synonyms, not embeddings.

B.1 Matching Hierarchy

METEOR performs matching in this order:

1. **Exact match:** Same word (case-insensitive)
   - “results” “results”
2. **Stem match:** Words sharing the same stem (Porter stemmer)
   - “produced” “producing” “produce”
3. **Synonym match:** Words in the same WordNet synset
   - “yield” “produce” “generate”
   - “results” “findings”

B.2 Example

- **Reference:** “The experiment yielded significant results.”
- **Model:** “The experiment produced significant findings.”

METEOR matches:

Word Pair	Match Type
yielded produced	Synonym
results findings	Synonym
significant significant	Exact

Result: High METEOR score despite different wording.

B.3 Comparison

Metric	Detects Synonyms?	Method
BLEU	No	N-gram overlap only
ROUGE	No	Lexical overlap only
METEOR	Yes	WordNet + stemming
BERTScore	Yes	Embedding similarity

Limitation: METEOR is dictionary-based, so it may miss domain-specific synonyms not in WordNet.

Appendix C: ROUGE-L and Word Order

ROUGE-L uses **Longest Common Subsequence (LCS)**, which allows flexible word ordering but requires preserved *relative* order.

C.1 How LCS Works

- Words don't need to be adjacent
- Words must appear in the same relative order
- Reversed order breaks the match

C.2 Examples

Order changed, relative order preserved (works):

- Reference: "Transformers model long-range dependencies."
- Candidate: "Long-range dependencies are modeled by transformers."
- LCS: "transformers → model → long-range → dependencies"
- Result: Good ROUGE-L score

Order reversed (fails):

- Reference: "A B C D"
- Candidate: "D C B A"
- LCS: Only 1 token
- Result: Bad ROUGE-L score

C.3 Metric Comparison

Metric	Enforces Adjacency?	Enforces Order?	Captures Paraphrase?
BLEU	Yes	Yes	No
ROUGE-L	No	Relative	Partial
METEOR	No	Yes	Synonyms
BERTScore	No	No	Best

C.4 Key Takeaway

ROUGE-L is more flexible than BLEU but not truly order-agnostic. It relaxes the adjacency constraint while still requiring relative order preservation.

References

1. Papineni, K., et al. (2002). "BLEU: a Method for Automatic Evaluation of Machine Translation"
2. Lin, C.-Y. (2004). "ROUGE: A Package for Automatic Evaluation of Summaries"
3. Banerjee, S., & Lavie, A. (2005). "METEOR: An Automatic Metric for MT Evaluation"
4. Zhang, T., et al. (2020). "BERTScore: Evaluating Text Generation with BERT"
5. Hendrycks, D., et al. (2021). "Measuring Massive Multitask Language Understanding" (MMLU)
6. Cobbe, K., et al. (2021). "Training Verifiers to Solve Math Word Problems" (GSM8K)