developments:
13 January 2026-20 January 2026

scope

- **Given:**
  a .xlsx file in which each row stores three pieces of data about a patient: a clinician report, an LLM-generated Alzheimer's disease progression score, and the LLM's reasoning for the score

  example (not real data… made up for the purpose of explanation only):

89 year old female patient The patient has faced memory loss for the past three years. Alzheimer's disease biomarker observed. Severe volume loss and hippocampal atrophy. - Dr. John Doe at Big Brain Hospital Neuroscience Department	7	- history of memory loss - Alzheimer's biomarker - atrophy

- The **goal** is to produce a numerical metric for how accurate the LLM-generated score is by analyzing how closely the LLM's reasoning and the real clinician's report align.

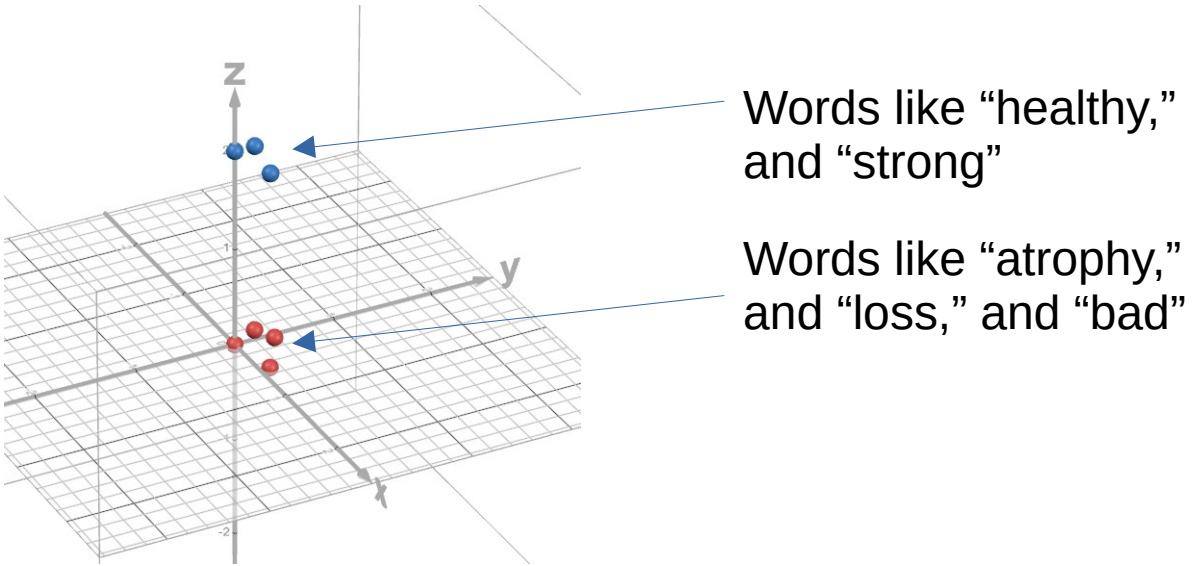
BERTScore

- **BERTScore** is a metric that measures the similarity between two different pieces of texts (i.e., an LLM-generated piece of text and a base truth piece of text).
- How is a BERTScore metric generated?

1) Both the LLM-generated text and the base truth text are **tokenized** (broken up into granular units of text like words).
2) Then, an **embedding** is generated for each token in both pieces of text. An embedding is a vector representation of a token.
3) To determine similarity, comparison of cosines (i.e., cosine similarity) of the embeddings (which are vectors) is performed.

BERTScore: Embeddings

- The process of embedding maps a token (like a word or any other basic text unit) to a vector of floating-point numbers.
  - The idea is that semantics will be embodied because vectors for tokens of similar meanings should be relatively close to each other. Vectors for tokens of say, the opposite meaning will be far away.



- BERT, a language model by Google, is used to generate embeddings from tokens.

BERTScore: Cosine Similarity

- The cosine similarity is computed for the embedding of each token of the LLM-generated text with the embedding of each token of the base truth text.
  - Recall that embeddings are vectors of floating-point numbers. The angle between any two vectors can be used to quantify how “similar” or “close” they are. In particular, the cosine of this angle is being studied. Greedy matching is used to select the highest cosine similarity score for each token.
  - The highest cosine similarity scores are then used to compute metrics like BERTRecall and BERTPrecision.

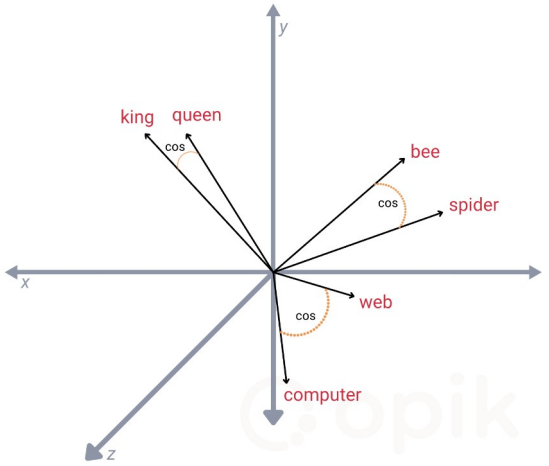


image courtesy of Opik (https://www.comet.com/site/wp-content/uploads/2024/12/Screenshot-2024-12-17-at-8.10.34%E2%80%AFPM.png)

BERTScore: BERTPrecision

- “quantifies how much of the candidate's content is semantically meaningful relative to the reference” – Abby Morgan (Comet)

$$P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j$$

image courtesy of Comet: https://www.comet.com/site/wp-content/uploads/2024/12/Screenshot-2024-12-14-at-8.05.06%E2%80%AFPM-1024x443.png

BERTScore: BERTRecall

- “BERTRecall reflects how much of the reference’s meaning is captured by the candidate.” – Abby Morgan (Comet)

$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j$$

image courtesy of Comet: https://www.comet.com/site/wp-content/uploads/2024/12/Screenshot-2024-12-14-at-8.04.30%E2%80%AFPM-1536x609.png

BERTScore Literature

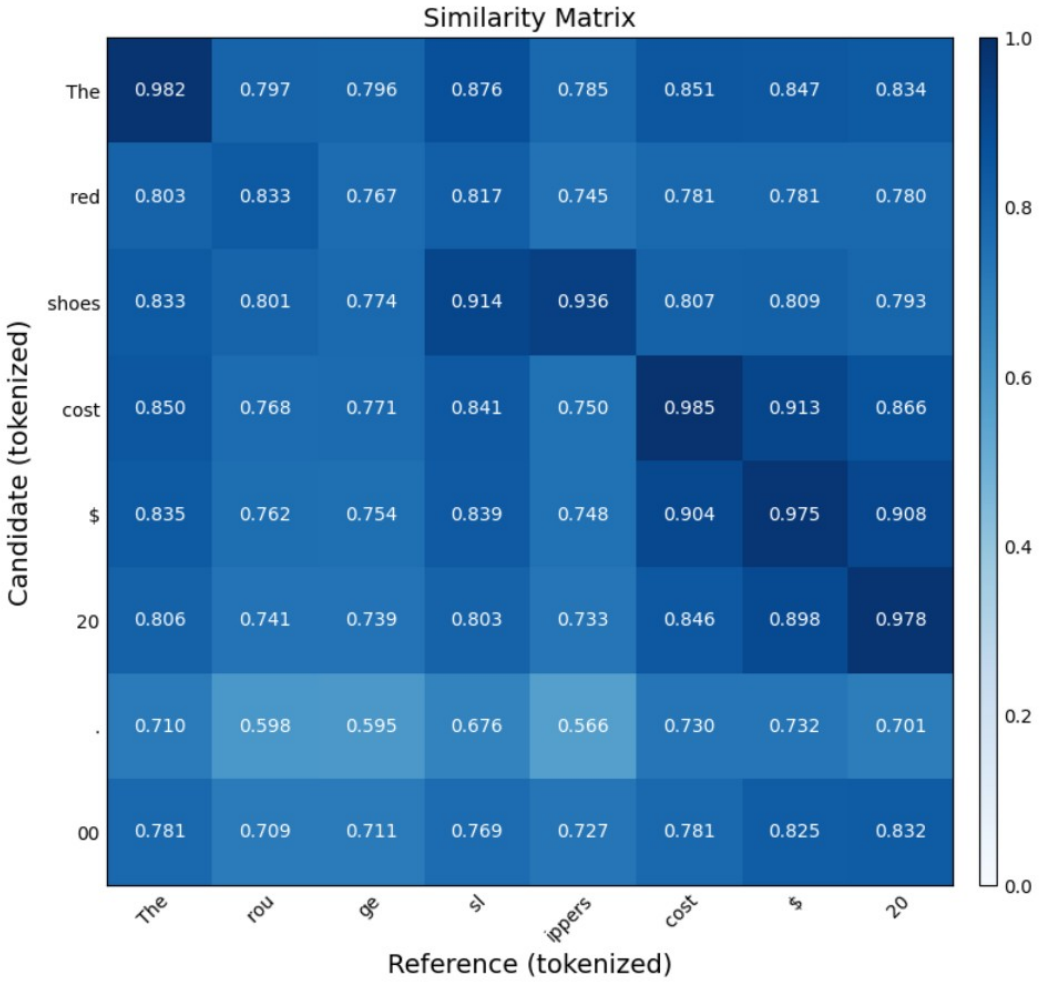
- Liang, D. (2024, July 30). Intro — Getting Started with Text Embeddings: Using BERT. Medium. https://medium.com/@davidlfliang/intro-getting-started-with-text-embeddings-using-bert-9f8c3b98dee6
- Morgan, A. (2024, December 19). BERTScore For LLM Evaluation. Comet. https://www.comet.com/site/blog/bertscore-for-llm-evaluation/

developments:
20 January 2026-27 January 2026

Clarifying What is Being Done w/ the Cosine Similarity Scores

- Both the candidate and reference texts are tokenized.
- The cosine similarity is calculated between each token in the candidate text and each token in the reference text.

image courtesy of Comet: https://www.comet.com/site/wp-content/uploads/2024/12/similarity_matrix.png



$$P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j$$

image courtesy of Comet: https://www.comet.com/site/wp-content/uploads/2024/12/Screenshot-2024-12-14-at-8.05.06%E2%80%AFPM-1536x665.png

INTERPRETATION:
For each candidate token embedding, find the reference token embedding that is the most similar (highest cosine similarity score).
Add that maximum cosine similarity score to a running sum, and divide that sum by the number of candidate tokens for an average.

ROUGE: Let's compare BERTScore to a more "rigid" evaluation metric

- There are many different types of ROUGE metrics; some include ROUGE-1, ROUGE-2, and ROUGE-L, ROGUE-S, and ROGUE-SU.
  - What's the difference?
    - Each compares an LLM summary to human reference text at different granularity.
- ROUGE-1: precision of single-word overlap between candidate and reference texts

$$\text{Rouge} - 1\ (\text{Recall}) = \frac{unigram\ matches}{unigram\ in\ reference}$$

Rouge-1 (Recall), Source: Author

$$\text{Rouge} - 1\ (\text{Precision}) = \frac{unigram\ matches}{unigram\ in\ output}$$

Rouge-1 (Precision), Source: Author

$$\text{Rouge} - 1\ (\text{F1}) = 2\ \text{x}\ \frac{Precision\ x\ Recall}{Precision + Recall}$$

Rouge-1 (F1), Source: Author

Image courtesy of Muhammad Umar Amanat: https://miro.medium.com/v2/resize:fit:828/format:webp/1*eV_qPGSS06YZbbUUI1tJcw.png

- ROUGE-2 is like ROUGE-1 except using bigrams instead of unigrams
- ROUGE-L uses the LCS (longest common subsequence) between the candidate and reference texts.

$$\text{Rouge} - \text{L (Recall)} = \frac{Length\ of\ LCS}{unigrams\ in\ reference}$$

Rouge-L (Recall), Source: Author

$$\text{Rouge} - \text{L (Precision)} = \frac{Length\ of\ LCS}{unigrams\ in\ output}$$

Rouge-L (Precision), Source: Author

$$\text{Rouge} - \text{L (Precision)} = 2\ \text{x}\ \frac{Precision\ x\ Recall}{Precision + Recall}$$

Rouge-L (F1), Source: Author

Image courtesy of Muhammad Umar Amanat: https://miro.medium.com/v2/resize:fit:828/format:webp/1*p3rm7-4J49uE6CGnYH_jmQ.png

ROUGE-1 Example

- Candidate: {I, like, to, eat, apples}
- Reference: {I, like, apples, a, lot}
- Overlap: {I, like, apples}
- **Precision: 3/5**
- **Recall: 3/5**
- **F-1: 3/5**

ROUGE Literature

Amanat, Muhammad Umar. “LLM Evaluation with Rouge.” Medium, 19 Jan. 2024, medium.com/@MUmarAmanat/llm-evaluation-with-rouge-0ebf6cf2aed4.

Gupta, Mehul. “LLM Evaluation Metrics Explained - Data Science in Your Pocket - Medium.” Medium, Data Science in your pocket, 19 June 2024, medium.com/data-science-in-your-pocket/llm-evaluation-metrics-explained-af14f26536d2.

I will use these metrics...

- BERTScore
- ROUGE-1, ROUGE-2, ROUGE-L

Note:

ROUGE scores are between 0 and 1 (higher is better).

When I tested two pieces of exact match texts for BERTScore, I yielded 1.

Proxy (“fake data”) Test Case Sets

- **Highly accurate** (candidates list reasons that are actually in the reference)
- **Very Inaccurate** (candidates hallucinate reasons that are not discussed in the reference)
- **False Positives** (reference says that a symptom/condition is not present, but the LLM misunderstands and declares a false positive)