What a G-Eval score represents

- **Normalized quality score (0–1 range):** DeepEval's G-Eval returns a floating score from 0 to 1, where higher is better [1]. By default, 1.0 means "perfect" and 0.0 means "completely incorrect." A score of 0.88 thus indicates the answer is highly close to the ideal, whereas scores near 0 (e.g. 0.02) indicate a strong failure [2] [3].
- **Pass/fail threshold:** DeepEval uses a default pass threshold of 0.5 [1]. 0.88 is well above this threshold, so in practice the answer "passes" the quality criteria. In other words, the LLM judge found it largely correct/coherent. For context, one tutorial example gave 1.0 for an exact match, ~0.7 when details were missing, and ~0.02 for a direct contradiction [2] [3]. By that standard, 0.88 implies very few errors (likely only minor omissions).
- **Criteria-dependent:** The raw meaning of the score depends on the **evaluation criteria** you set. In a healthcare Q&A, criteria might include *factual correctness* (no medical errors), *relevance* (answering the question), *coherence* (clear reasoning), and *groundedness* (supported by reference data). The G-Eval score indicates how well the answer met these criteria [4] [5]. For example, one can define steps like "Check if any facts in the answer contradict the trusted medical reference" or "Does the answer omit critical steps?" [6] [5]. The higher the score, the better the answer satisfied those checks.

How G-Eval scores are computed

- **LLM-as-judge with Chain-of-Thought:** G-Eval treats a strong LLM (GPT-4 by default) as a judge that reasons through the answer. It is a two-stage process [7] [8]:
- **Generate evaluation steps (CoT):** Given your criteria (e.g. "verify factual accuracy and relevance"), G-Eval first has the LLM outline a chain of reasoning in natural language. These *evaluation steps* might include things like "Compare facts in the answer to the expected answer," "Penalize missing details," etc [7] [6]. (You can provide your own steps or let G-Eval generate them from the criteria.)
- **Score with form-filling:** The evaluation steps are concatenated with the question, the generated answer (actual_output), and the ground-truth answer (expected_output). The LLM is then prompted to output a **rating** (typically 1–5) based on those steps [8]. In practice, DeepEval asks the LLM to "score the answer from 1 to 5" given the reasoning steps.
- **Normalization:** That raw 1–5 rating is converted into a 0–1 score. DeepEval does this by looking at the output token probabilities of the LLM and computing a weighted sum, which mitigates bias in the LLM's generation [8]. The end result is a continuous G-Eval score (0–1) that reflects the weighted average rating.
- **Evaluation parameters:** In your test case you supply parameters like the question, actual answer, expected (ground-truth) answer, and any context or retrieved documents. G-Eval will compare the actual answer against the expected output (and context, if provided) according to the steps. For example, you might instruct: "Check whether the facts in the actual answer contradict the reference answer; penalize omission of detail; vague phrasing is OK" [6]. The LLM then uses these to judge factuality, relevance, coherence, etc. (Additional metrics like "Answer Relevancy" or "Faithfulness" exist in DeepEval if you want more targeted checks.)

- **Optional rubric and strict mode:** You can constrain the scoring by providing a rubric or using strict_mode. By default G-Eval outputs a graded score. For instance, DeepEval documentation shows rubrics mapping score ranges to labels (e.g. 7–9/10 = "correct but missing minor details") [9]. With strict_mode on, the metric would only give 1.0 or 0.0 (pass/fail). But typically G-Eval is used in its graded form so you see variations like 0.88.

Comparing multiple answers per question

- **Independent scoring:** Each generated answer is evaluated **separately** with G-Eval. Thus for one question you'll get three independent scores (one per answer). There is no built-in normalization across them – each score shows that answer's quality per the criteria.
- **Ranking and thresholding:** You can rank the answers by score to pick the best one. The answer with the highest G-Eval score is judged most aligned with the criteria. For example, if answers scored 0.88, 0.75, and 0.60, you would consider the 0.88-answer the strongest. All three here are above 0.5 (passing), but 0.88 indicates a much higher quality. If instead one answer scored below 0.5 (say 0.30), you'd mark that answer as failing the criteria and discard it. In the DataCamp tutorial, 3 answers got scores 1.0, 0.7, and ~0.02: two passed ( ) and one failed ( ) [2] [3]. That 0.02 answer was contradictory, the 0.7 answer missed some details. Similarly, you would interpret your three scores to see which answers pass and how strong each is.
- **Interpreting score gaps:** Large gaps are meaningful. An answer at 0.88 vs another at 0.50 suggests the first is far better. The example above (66.7% pass rate) had one perfect answer, one partially correct, one wrong [10]. The reasons given by G-Eval ("perfect match" vs "missing details" vs "contradiction") help explain the scores. You can use these "reason" texts (DeepEval returns them alongside scores) to understand *why* one answer scored higher than another.

Is 0.88 "good"? Benchmarks and context

- **Above average:** Because 0.5 is the pass cut-off, a score like 0.88 is generally considered **high** in DeepEval. It indicates the answer met most criteria. For perspective, an example score of 0.7 was described as "the main idea is present but details omitted" [11]. A 0.88 would imply even fewer omissions or errors. In DeepEval's optional rubric (0–10 scale), 7–9 corresponds to "correct but missing minor details" [9]. A raw 0.88 ($\approx$8.8/10) falls well into that upper band. So we'd call 0.88 a *good to excellent* score.
- **No absolute "100%":** Keep in mind G-Eval is probabilistic. Scores aren't like human-graded percentages exactly, and there's no fixed "passing grade" beyond threshold. But in practice, models and evaluators often treat $\geq$0.8–0.9 as very strong performance. A score near 1.0 means almost perfect alignment to the expected answer [2].
- **Model bias caveat:** Note that LLM-based evals can have biases (e.g. slight preference for LLM-generated text, as the G-Eval paper notes [12]). In general, treat scores as a heuristic. But 0.88 is well above threshold, so it's safe to say the answer is substantially correct under the chosen criteria.

Using G-Eval scores to assess model performance

- **Per-question decisions:** You can use the scores directly to choose an answer per question. For instance, select the answer with the highest score (if $\geq$0.5) as your "best" answer. If all answers score poorly, you may flag that question for model improvement.

- **Aggregate metrics:** Over many questions (e.g. your 50-question dataset), you can compute summary statistics. Common choices are the *average G-Eval score* or the *pass rate* (fraction of answers with score ≥ threshold) [10]. For example, if 40 of 50 answers score ≥0.5, the pass rate is 80%. The DataCamp guide explicitly reports that "3 test cases…achieved a 66.7% pass rate" when 2 of 3 passed [10]. You can do the same at scale to quantify model accuracy on the criteria.
- **Comparing models or prompts:** If you have multiple models or prompting strategies, compare their average or pass-rate G-Eval scores. A higher average score (or higher % of answers above, say, 0.8) indicates a stronger model under the defined criteria. You can also perform statistical tests on the score distributions.
- **Error analysis:** DeepEval returns qualitative "reason" strings explaining each score. Use these for debugging. For example, if many answers score ~0.6 with reasons like "missing details," you know to focus on completeness. If many hit low scores citing contradictions, you need to improve factual accuracy.
- **Setting quality thresholds:** In a clinical context, you might impose stricter standards. E.g. require G-Eval ≥0.7 for "publishable" answers, or trigger a human-review if the score is below 0.5. Because G-Eval is designed to align with human judgment [12], you can treat these numeric thresholds as proxies for answer quality standards.

G-Eval vs. other evaluation methods

- **Traditional metrics (BLEU/ROUGE/etc.):** These rely on n-gram overlap and often fail to capture factual correctness or reasoning quality [12]. They also need reference text. G-Eval, in contrast, uses an LLM to judge meaning and logic, so it can catch errors or omissions that BLEU/ROUGE would miss. (Indeed, the G-Eval paper shows much higher correlation with human judgments than BLEU/ROUGE [12].)
- **Embedding/textual similarity metrics:** Tools like BERTScore or cosine similarity measure "surface" similarity, but not whether the answer actually answers the question or is factually right. G-Eval explicitly checks the content against the expected answer, making it more reliable for QA.
- **LLM-based evaluators:** G-Eval is part of a class of LLM-as-judge methods (similar in spirit to OpenAI's Evals or other GPT-4 scoring). Its distinguishing feature is the built-in chain-of-thought "form-filling" approach [8] and deep customization via criteria. It is essentially an open-source, highly configurable version of an LLM evaluation. For instance, OpenAI Evals also prompt a model to rate answers, but DeepEval wraps this in a reusable framework with metric objects and test cases [4].
- **DAG (DeepEval's other metric):** DeepEval also offers a "DAG" metric (decision-tree logic). DAG is deterministic and rule-based, while G-Eval is generative. The DeepEval docs recommend G-Eval for "subjective" criteria like correctness or coherence and DAG for strict format checks [13]. In practice, G-Eval is easier to set up ("takes no effort") and handles nuance better [13], whereas DAG gives more predictable but rigid results.
- **Human evaluation:** Manual human judgment is the gold standard but is expensive and variable. G-Eval aims to approximate human judgments at scale. The original G-Eval study found that GPT-4's scores had notably higher agreement with humans than older automatic metrics [12]. In a healthcare QA pipeline, using G-Eval means you can automatically assess thousands of answers with near-human quality of judgment.
- **Summary:** G-Eval (in DeepEval) is best seen as an **LLM-empowered quality check**. It fills the gap between crude overlap metrics and costly human review. By leveraging GPT's reasoning, it captures nuance (factuality, coherence, relevance) that simple metrics miss [4] [12]. In your use case, it

complements domain-specific checks (e.g. making sure medical terms are correct) and provides a quantitative score you can use to rank or filter answers.

Sources: DeepEval's documentation and tutorials describe G-Eval as an LLM-based scoring metric using chain-of-thought [4] [8]. In practice, scores reflect how well an answer matches the ground truth under custom criteria (such as factual accuracy, relevance, etc.), normalized to [0,1] [1] [2]. G-Eval (GPT-4) has been shown to align more closely with human judgments than traditional metrics [12], making it a powerful tool for evaluating healthcare QA outputs.

---

[1] [4] [8] [9] G-Eval | DeepEval - The Open-Source LLM Evaluation Framework
https://docs.confident-ai.com/docs/metrics-llm-evals

[2] [3] [6] [10] [11] Evaluate LLMs Effectively Using DeepEval: A Practical Guide | DataCamp
https://www.datacamp.com/tutorial/deepeval

[5] [13] Introduction | DeepEval - The Open-Source LLM Evaluation Framework
https://docs.confident-ai.com/docs/metrics-introduction

[7] Effective LLM Assessment with DeepEval
https://www.analyticsvidhya.com/blog/2025/01/llm-assessment-with-deepeval/

[12] G-Eval: NLG Evaluation using Gpt-4 with Better Human Alignment - ACL Anthology
https://aclanthology.org/2023.emnlp-main.153/