Improving Medical Hallucination Detection with System Combination and Rule-based Customization

Jonathan Lasko, Damianos Karakos, Francis Keith
RTX BBN Technologies
Cambridge, MA
{jonathan.lasko,damianos.karakos,francis.keith}@rtx.com

Abstract

The presence of factuality errors (hallucinations) in the outputs of patient-facing medical chatbots is a serious problem: they can lead to patient harm and erode people's trust in the medical profession. For this reason, it is crucial to detect hallucinations in chatbot outputs and forward them to clinicians for review. In this paper, we present the system we built for detecting such errors: it consists of multiple LLM-powered detectors which are combined together with a novel alignment procedure. We ran our system on the MedExpert-Benchmark dataset (Yarmohammadi et al., 2025) and our results on two use cases, Mental Health and Prenatal Care, show that the combined system gives nice gains over the individual systems. Additionally, we show that further customization of the system to each one of the use cases leads to further gains, but at the cost of reduced generalizability. Our code and dataset are available here: https://github.com/BBN-E/medic-customnlp4u.

1 Introduction

It is generally accepted that patient-facing medical chatbots have the potential to transform the practice of medicine at many levels: more widespread and faster access to healthcare, supporting patients in understanding their condition and managing their symptoms, display of empathy and stronger guarantees on privacy. However, the presence of factuality errors (hallucinations) in the outputs of medical chatbots is still a significant problem, as they can lead the patient the wrong path in terms of diagnosis and treatment. Clearly, such errors can result in patient harm, ranging (in terms of severity) from minor to life-threatening.

In this paper, we present a hallucination detection system, consisting of multiple components which are combined together. The components consist of LLM-as-judge, RAG, and Agentic workflows, and they all "vote" on the presence of a hallucination in the medical answer. The confidence of the detector, along with the estimated harm level and the automatically-generated explanation, are used as input into a novel combination scheme, which *aligns* the individual detections and combines their strengths, leading to improved performance. It is important to note that, in our setup, the error detectors can only *observe* the output of the chatbot; they cannot probe the chatbot by prompting it with variations of the question, or perturb its generations in some other way, or access its internal components. Although such perturbation approaches have been found to be successful for estimating the confidence of the chatbot (and, hence, its tendency to produce errors) (Manakul et al., 2023) they are not relevant in our case; the chatbot's response is all we have available.

Additionally, we perform an analysis of the system's failure modes that are due to non-medical reasons. We then use this analysis to come up with rules that customize the evaluation technology on each of the use cases under consideration. Application of the rules leads to improved performance, but at the cost of reduced generalizability.

2 Prior Work

Hallucination detection in the outputs of LLMs (and medical chatbots) has been an active area of research. For general hallucination detection, a popular approach (e.g., of (Manakul et al., 2023)) is that of generating multiple LLM outputs and then comparing them with each other to measure stability. However, as mentioned earlier, our setup does not allow us to generate multiple chatbot outputs per answer; one output is all we have. This also rules out white-box methods, which assume that likelihoods or other internal model components are observable. Multiple other approaches (e.g., (Agarwal et al., 2024; Pandit et al., 2025)) use LLM-as-judge for automatic hallucination detec-

tion, but these detections are not combined together. Although there are approaches for combining together hallucination detections such as (Chen and Mueller, 2024; Bouchard and Chauhan, 2025), they are not applied to the medical domain. A notable exception is the very recent paper by (Hussain et al., 2026), which uses machine learning on top of LLM-derived features. This approach is very different from the one we describe in this paper; we plan to compare the two in a future publication.

3 The Detection System

Our hallucination detection system takes as input a medical chatbot response and routes it through multiple detection components, which can potentially generate multiple detections per response. The outputs of these components are combined together using an alignment-based system combination approach, similar to how it is done in other fields such as Speech Recognition and Machine Translation (Fiscus, 1997; Rosti et al., 2008). The various system components are briefly described below.

3.1 Detection Components

LLM-as-Judge: We used a similar approach as in (Liu et al., 2025). Briefly, the LLM-as-judge detector utilizes the internal knowledge of the LLM to find hallucinations, using a prompt similar to the one shown in Section 10.9.1 of (Liu et al., 2025). Both open-source LLMs (e.g., Llama3-ChatQA-1.5-8b) and closed, API-based LLMs (e.g., GPT-5) are supported. To find hallucinations, the detector prompts an LLM to find sentences that contain hallucinations and outputs (i) each sentence containing a detected hallucination; (ii) an explanation for why this is a hallucination; and (iii) an estimated level of harm to the patient (chosen from "none", "very low", "low", "medium", "high"). The LLM-as-judge Detector loads open-source LLMs via the Hugging Face transformers API (Hugging Face) and utilizes langchain (LangChain) to solicit and parse structured responses.

LLM-as-Judge with RAG: Our RAG-based LLM-as-Judge hallucination detector (or, more simply, RAG Detector) uses retrieval-augmented generation (RAG) to augment the information an LLM can utilize while evaluating an answer for the presence of hallucinations. Specifically, the RAG Detector takes the original question and uses it with a retriever module optimized for the medical domain to query a database of high-quality medical reference data (e.g., DSM-V and other medical textbooks, and the StatPearls medical test practice material). The corpora are taken from MedRAG (Xiong et al., 2024). Like the LLM-as-judge Detector, the RAG Detector uses the Hugging Face transformers and langchain APIs.

Agentic: Our agentic hallucination detector follows an approach similar to (Liu et al., 2025). It places hallucination detection agents in a semi-structured round-robin conversation with critic agents who point out various weaknesses in the detections. Each agent is assigned its task via user-provided system prompts, including instructions to use JSON formats for the detections. A reviewer agent supervises the conversation, instructing the detector agents to revise their detections based on feedback from critic agents, and determining when the revisions have converged. The conversation is terminated after a maximum number of rounds (30) are reached. The Harm Critic agent gives feedback on the patient harm levels assigned to each hallucination. The Explanation critic gives feedback on the explanations given for each detection by leveraging retrieval-augmented generation (RAG) from medical-specific text corpora for its critiques. For the system prompts used in the agentic detector, see Section 10.9.2 of (Liu et al., 2025).

MedScore: We used the two-step factuality detection system from (Huang et al., 2025). It first decomposes a sentence of a response into multiple claims (i.e., facts, statements) and then verifies each claim by a given gold reference (e.g. related medical snippets from PubMed, StatPearls, Medical Textbook). The decomposition step utilizes OpenAI's gpt-4o-mini (Hurst et al., 2024) to find context information and decomposes the targeted sentence into independent, condition-aware claims. In the second step, MedScore evaluates the correctness of each claim by prompting a verifier LLM (OpenAI's GPT-4o (Hurst et al., 2024)) to judge the claim's correctness using its internal knowledge. Optionally, the verifier can be configured to use the claim as a query to search for the top ten relevant medical snippets from the MedRAG corpus (Xiong et al., 2024); the verifier LLM then uses these snippets as a gold reference/context.

3.2 System Combination

As mentioned earlier, each hallucination detector outputs one or more detections per answer. In our approach to system combination, we aim at increasing the confidence of the detections that agree

with each other and (conversely) reducing the confidence of detections that do not agree with each other. To measure agreement, we proceed incrementally, by aligning the outputs of two detectors at a time: we first compute the cosine similarity between the embeddings of the explanations given by two detectors, and then use the Hungarian algorithm (Wikipedia contributors, n.d.) to align the two sets of detections together. If two detections are aligned, they are "merged" into one, and its score/confidence is set to the linear combination of the individual confidences. If a detection is not aligned with any other detection then its confidence is essentially interpolated with zero, thus, reducing its strength (discounted).

4 Experiments and Results

We ran our detection experiments on the publicly available dataset MedExpert-Benchmark (Yarmohammadi et al., 2025). It consists of two medical use cases, Mental Health (MH) and Prenatal Care (PC), consisting of 280 and 260 chatbot answers, respectively. The hallucination rates are: 27.5% for MH and 26.8% for PC. We measure performance using the F1 metric.

Results with the aforementioned detectors and with the system combinations appear in Table 1. In the case of MedScore (which uses GPT-4o) and LLM-as-a-judge with GPT-5, each run produces different results due to the random sampling done by these LLMs. For this reason, Table 1 shows the *average* precision, recall, and F1 across multiple runs of these two systems – four for MedScore, and five for LLM-as-a-judge with GPT-5. All individual samples are included in the system combinations.

As can be seen from these results, performance in terms of precision and recall can vary widely as a function of the approach and the underlying model. For example, for Mental Health, although their F1 scores are almost identical, GPT-5 as LLM-as-judge results in the best precision (38%) and low recall (63.4%) while the system combination without GPT-5 has a lower precision (31.8%) and a very high recall (81.8%). The system combination we propose does seem to combine the strengths of the different systems: combining the two aforementioned systems results in the best F1 score (51.9% for Mental Health, 51.8% for Prenatal Care) without using any complex machine learning system, resulting in both improved precision and recall. Interestingly, the combinations without GPT-5 result in F1 that is on par with GPT-5 alone.

5 Error Analysis

We looked at some of the errors made by our system on our MedExpert-Benchmark (Yarmohammadi et al., 2025) dataset. We focused on the *non-medical errors* made by LLM-based detector(s). These are errors caused by non-adherence to the provided guidelines, or due to a misunderstanding of the chatbot answer, or structural issues. The errors with a medical basis were forwarded to a clinician on our team; his analysis will appear in future work.

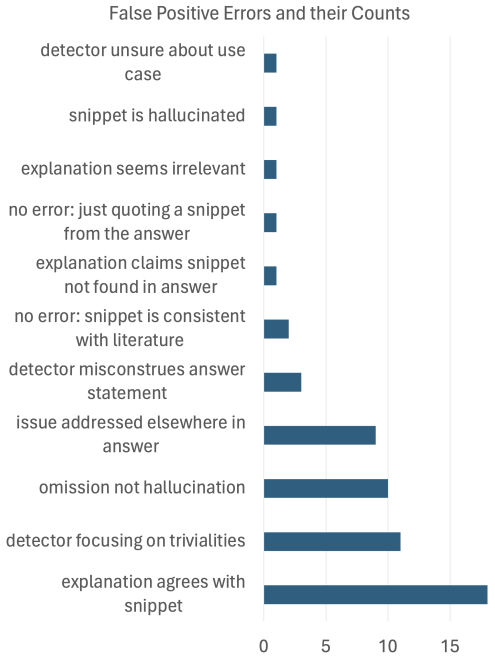

Figure 1: Breakdown of false positive detections of non-medical nature, based on our team's manual review. They are sorted from least frequent to most frequent).

We manually reviewed 192 hallucination detections made by our system over our two use cases. We focused mainly on false positives, as these are the ones affecting our precision score. We found out that 61 of these false positives (about 31.8%) did not seem to have a relevant medical reason. Figure 1 shows a breakdown of the identified error patterns over both use cases. Explanations of the most frequent of these patterns appear in Table 4 in the Appendix.

6 Customization and Compromises

In this section, we try to answer the question: how can we quickly improve performance on

Method	Model(s)	Mental Health			Prenatal Care		
		F1	Precision	Recall	F1	Precision	Recall
System Combination	Llama3-ChatQA-1.5-8B, Mistral-Nemo-Instruct-2407, GPT-4o-mini, GPT-4o, GPT-5	**0.519**	0.368	0.883	**0.518**	**0.402**	0.729
System Combination (no GPT-5)	Llama3-ChatQA-1.5-8B, Mistral-Nemo-Instruct-2407, GPT-4o-mini, GPT-4o	0.458	0.318	0.818	0.488	0.343	0.843
LLM-as-Judge	Llama3-ChatQA-1.5-8B	0.435	0.282	0.948	0.405	0.276	0.757
LLM-as-Judge*	GPT-5	0.466	**0.380**	0.634	0.486	0.361	0.786
RAG	Llama3-ChatQA-1.5-8B	0.445	0.286	**1.000**	0.432	0.279	0.957
Agentic	Mistral-Nemo-Instruct-2407	0.407	0.309	0.597	0.435	0.297	0.814
MedScore*	GPT-4o-mini, GPT-4o	0.372	0.279	0.562	0.465	0.324	0.825

Table 1: Hallucination detection results on MedExpert-Benchmark. The best result per use case is in **bold**. Rows for methods utilizing GPT-4 or GPT-5 are denoted with an asterisk (*) indicating that the Precision, Recall and F1 score are averaged across multiple runs to account for non-deterministic variations in the responses. LLM-as-Judge GPT-5 scores are averaged across 5 sample runs whose individual F1 scores ranged between 0.434 – 0.506 for MH and 0.477 - 0.508 for PC. MedScore scores are averaged across 4 sample runs whose individual F1 scores ranged between 0.361 - 0.395 for MH and between 0.444 - 0.488 for PC. Because of this, combining those rows' averaged precision and recall with the F1 formula does not result in the averaged F1 score we report in that row.

Use case	Rules	Original F1	New F1 after applying rules	New F1 after applying rules of other use case
Mental Health	Snippet says talk to a medical professional (MH) + Snippet Resembles Explanation (Jaccard) + Snippet Resembles Explanation (Levenshtein)	0.519	**0.540**	0.475
Prenatal Care	Snippet says talk to a medical professional (PC) + Snippet Resembles Explanation (Jaccard) + Snippet Resembles Explanation (Levenshtein) + Explanation is a critique of risk framing + Explanation references guidelines	0.518	**0.536**	0.515

Table 2: The impact of applying error-correcting rules to filter out false positives is approximately 2% absolute increase in F1 score for each use case.

MedExpert-Benchmark by customizing the detection system on the use cases of interest, and what compromises are required? The customization was done by creating sets of rules that address some of the types of errors we found in our analysis: the rules remove some of the erroneous detections made by our system.

Using the results from the error analysis described in Section 5 (where we manually reviewed false positives from our detectors against the MedExpert dataset), we crafted rules for filtering out false positives. Some of these initial rules for eliminating false positives (snippet/explanation similarity, omission phrases) showed promise. We then prompted GPT-5-thinking with two files, the annotated false positives and the full set of detections, for each use case, and instructed the LLM to identify patterns differentiating false positives from true positives. This produced an additional eight rules. Table 5 in the Appendix has the full set of rules.

Using these rules to filter out positives achieves approximately 2% absolute improvement in F1 score, for each use case, as shown in Table 2. Since each rule was derived from the outputs of our system using a single use case of the MedExpert dataset, they are a form of customization to the use case. Some rules may generalize across use cases, but others utilize domain-specific terminology (e.g., therapist, DSM, depression, serotonin). The best results for each use case were achieved using *different* rule sets, and using the rule set from one use case degraded performance on the other use case, as shown in the last column of Table 2. This means that these rules result in some kind of overfitting to the use cases and to specific patterns in our system's outputs.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	—	0.107	**0.0146**	0.0998	0.0988	0.6308	0.188	0.109	0.0842	**0.0006**	**0.0044**	0.4874	**0.0272**	**0.0**	**0.0**
2	0.0996	—	0.2336	**0.0088**	**0.0128**	0.129	**0.0224**	0.4118	0.5442	**0.0284**	**0.0324**	0.9862	0.227	**0.0**	**0.0**
3	0.5706	0.16	—	**0.0016**	**0.0032**	0.0556	**0.0066**	0.6256	0.8732	0.0754	0.0636	0.736	0.4502	**0.0**	**0.0**
4	0.83	0.155	0.5924	—	0.8776	0.0792	0.6602	**0.005**	**0.0046**	**0.0002**	**0.0002**	0.0668	**0.003**	**0.0**	**0.0**
5	0.0838	**0.0078**	**0.0352**	**0.0468**	—	0.1666	0.8172	**0.0086**	**0.0056**	**0.0004**	**0.0002**	0.0644	**0.0064**	**0.0**	**0.0**
6	0.296	**0.0362**	0.156	0.297	0.3846	—	0.2568	0.0652	0.0806	**0.0042**	**0.0014**	0.3202	0.054	**0.0002**	**0.0**
7	**0.0108**	**0.0006**	**0.0024**	**0.0044**	0.2366	**0.0446**	—	**0.0098**	**0.0108**	**0.0004**	**0.0002**	0.1212	**0.007**	**0.0**	**0.0**
8	**0.0**	**0.0**	**0.0**	**0.0028**	0.068	**0.0224**	0.3378	—	0.6076	0.613	0.07	0.3416	0.9268	0.0924	**0.0264**
9	**0.0286**	**0.0026**	**0.0104**	0.1568	0.6536	0.3594	0.8294	0.1956	—	**0.0378**	**0.0152**	0.6186	0.7424	**0.0014**	**0.0002**
10	**0.0466**	**0.0018**	**0.0146**	0.1928	0.7772	0.4178	0.6566	0.1544	0.8508	—	0.3094	0.1458	0.4574	**0.0276**	**0.0008**
11	**0.0414**	**0.0032**	**0.0134**	0.2054	0.7564	0.4382	0.6808	0.1388	0.8562	1.0	—	**0.0148**	0.2158	0.66	0.2658
12	0.2414	**0.0362**	0.1786	0.289	0.7222	0.4964	0.886	0.4738	0.9688	0.8608	0.8654	—	0.5212	**0.0084**	**0.0026**
13	**0.0108**	**0.0006**	**0.0024**	**0.0044**	0.2366	**0.0446**	1.0	0.3378	0.8294	0.6566	0.6808	0.886	—	**0.0146**	**0.0012**
14	**0.0032**	**0.0002**	**0.0012**	**0.0004**	**0.04**	**0.0094**	0.1006	0.7266	0.1872	0.1312	0.1144	0.335	0.1006	—	**0.0008**
15	**0.0004**	**0.0002**	**0.0008**	**0.0002**	**0.014**	**0.001**	**0.0282**	0.3478	0.066	**0.043**	**0.0338**	0.1644	**0.0282**	0.169	—

Table 3: Combined pairwise p-value table. The upper-triangular entries correspond to Mental Health, the lower-triangular entries correspond to Prenatal Care. Boldface indicates statistical significance ($p \leq 0.05$).

7 Comparing Hallucination Detection Outputs

We used a paired bootstrap test on F1 (Berg-Kirkpatrick et al., 2012) to compare all pairs of hallucination detection outputs, including using the rules of Table 2. The p-values from all these comparisons are shown in Table 3; the upper-triangular entries correspond to MH and the lower-triangular entries correspond to PC (we combined the two use cases into a single table to save space). The rows (and columns) correspond to the following outputs:
row 1: Agentic
row 2: LLM-as-Judge (Llama3)
row 3: RAG
rows 4-7: four MedScore runs
rows 8-12: five LLM-as-Judge runs with GPT-5
row 13: System Combination without GPT-5
row 14: System Combination with GPT-5
row 15: after applying rules

Some observations are in order: (i) The differences between the "weakest" systems (low-index rows and columns) and the "strongest" systems (high-index rows and columns) are clearly statistically significant; these correspond to the table entries far from the main diagonal. (ii) The MH case has 56 pairs (53.3%) of outputs that have a statistically significant difference, while the PC case has 45 such pairs (42.9%); this could be a consequence of the fact that PC has fewer data points. (iii) The system combinations give statistically significant gains when compared to most of the single-system outputs that were included in the combination. However, the difference from the *best* single system included in the combination is not always statistically significant. (iv) Outputs obtained with the same "family" of systems via sampling (such as MedScore runs or LLM-as-Judge runs with GPT-5) do not always have a non-statistically significant difference; for example, half of the differences between MedScore runs are statistically significant in the PC case, while no MedScore differences are statistically significant in the MH case (rows/columns 4-7). We see the opposite trend in the case of LLM-as-Judge with GPT-5 (rows/columns 8-12).

8 Concluding Remarks

In this paper, we tried to answer a number of research questions related to the evaluation of the factuality of medical chatbots. We found that our detection system's main weakness is its low precision; it falsely detects too many hallucinations, despite the fact that system combination does improve precision and/or recall over the best individual system. Our error analysis of these false positives revealed that there are many cases in which the LLM-as-judge does not properly follow directions, misunderstands the task, or simply hallucinates about the presence of errors. Based on what we found, we came up with various rules (and we prompted GPT-5 for additional rules) that helped us reduce the prevalence of false positives. These rules are a form of customization of the system to the medical use case; using these rules on a different use case degraded results. We hope that this finding will help guide future research in this area.

9 Acknowledgments

This research was, in part, funded by the Advanced Research Projects Agency for Health (ARPA-H). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Government.

References

Vibhor Agarwal, Yiqiao Jin, Mohit Chandra, Munmun De Choudhury, Srijan Kumar, and Nishanth Sastry. 2024. Medhalu: Hallucinations in responses to healthcare queries by large language models. *arXiv preprint arXiv:2409.19492*.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 995–1005.

Dylan Bouchard and Mohit Singh Chauhan. 2025. Uncertainty quantification for language models: A suite of black-box, white-box, llm judge, and ensemble scorers. *arXiv preprint arXiv:2504.19254*.

Jiuhai Chen and Jonas Mueller. 2024. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5186–5200.

Jonathan G Fiscus. 1997. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover). In *1997 IEEE workshop on automatic speech recognition and understanding proceedings*, pages 347–354. IEEE.

Heyuan Huang, Alexandra DeLucia, Vijay Murari Tiyyala, and Mark Dredze. 2025. Medscore: Factuality evaluation of free-form medical answers. *arXiv e-prints*, pages arXiv–2505.

Hugging Face. Transformers. https://huggingface.co/docs/transformers/en/index. Online; accessed 2026-04-02.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Khizar Hussain, Bradley A Malin, Zhijun Yin, Susannah Leigh Rose, and Murat Kantarcioglu. 2026. Blending human and llm expertise to detect hallucinations and omissions in mental health chatbot responses. *arXiv preprint arXiv:2604.06216*.

LangChain. Langchain docs. https://docs.langchain.com/oss/python/langchain/overview. Online; accessed 2026-04-02.

Jonathan Liu, Haoling Qiu, Jonathan Lasko, Damianos Karakos, Mahsa Yarmohammadi, and Mark Dredze. 2025. Statistically significant results on biases and errors of llms do not guarantee generalizable results. *The Second Workshop on GenAI for Health Potential, Trust, and Policy Compliance (GenAI4Health@NeurIPS 2025)*.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 9004–9017.

Shrey Pandit, Jiawei Xu, Junyuan Hong, Zhangyang Wang, Tianlong Chen, Kaidi Xu, and Ying Ding. 2025. Medhallu: A comprehensive benchmark for detecting medical hallucinations in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2858–2873.

Antti-Veikko Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2008. Incremental hypothesis alignment for building confusion networks with application to machine translation system combination. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 183–186.

Wikipedia contributors. n.d. Hungarian algorithm — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Hungarian_algorithm. [Online; accessed 2026-04-02].

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6233–6251.

Mahsa Yarmohammadi, Alexandra DeLucia, Lillian C Chen, Leslie Miller, Heyuan Huang, Sonal Joshi, Jonathan Lasko, Sarah Collica, Ryan Moore, Haoling Qiu, and 1 others. 2025. Medexpert: An expert-annotated dataset for medical chatbot evaluation. In *Machine Learning for Health 2025*.

A Appendix

Error Category	Description
Explanation agrees with snippet	Detector copies or near-copies the snippet as its explanation
Detector focusing on trivialities	Snippet and explanation point to an insignificant issue
Omission not hallucination	Explanation flags a missing piece rather than a hallucinated claim
Issue addressed elsewhere in the answer	Detector ignores other parts of the answer that address the concern
Detector misconstrues answer statement	Detector misinterprets the original answer
Snippet is consistent with literature	Detector declares that snippet agrees with literature
Explanation claims snippet not found in answer	Detector wrongfully claims that snippet is not in the answer
Just quoting a snippet from the answer	Detector just copies a snippet from the answer
Explanation seems irrelevant	Explanation is not relevant to the snippet
Snippet is hallucinated	The snippet does not appear in the answer, even in approximate form
Detector unsure about use case	Detector gives explanation that is irrelevant to use case

Table 4: Description of the non-medical false-positive error categories

Use Case	Rule	Implementation Details (Key Phrases or Thresholds)	F1
MH	Snippet resembles Explanation (J)	Jaccard similarity between 3-grams from snippet and 3-grams from explanation	**0.524**
MH	Snippet resembles Explanation (L)	Normalized Levenshtein similarity score between snippet and explanation	**0.525**
MH	Snippet says talk to a medical professional	consult (a|your) (doctor|physician|professional|therapist)	**0.533**
MH	Snippet has non-medical hedging	may|might|can|could|often|sometimes|generally|typically BUT NOT mg|dose|dosage|treatment|therapy|diagnosis|diagnostic|schizophrenia| depression|bipolar|anxiety|melatonin|serotonin|ssri|snri|antidepressant| antipsychotic| benzodiazepine|symptom|disorder|disease	0.504
MH	Snippet has psycho-education	can affect|can influence|can impact|often involves|may experience|may develop|may have symptoms|can lead to (symptoms|changes)|can cause (symptoms|changes)|is often associated with|may include symptoms	0.509
MH	Explanation supports Snippet	is consistent with	**0.521**
MH	Explanation contains omission phrases	((important|crucial|essential|fails) to|does not|doesn't|omits|lacks) (mention|state|note|emphasize|highlight)	0.490
MH	Explanation references guidelines	guideline|guidelines|current evidence|literature|DSM|ICD|criteria	0.500
MH	Explanation overly long	Threshold at 55 words	0.500
MH	Explanation critiques framing	may mislead patients|create barriers|framing|overemphasizing	**0.520**
PC	Snippet resembles Explanation (J)	Jaccard similarity between 3-grams from snippet and 3-grams from explanation	**0.523**
PC	Snippet resembles Explanation (L)	Normalized Levenshtein similarity score between snippet and explanation	**0.520**
PC	Snippet says talk to a medical professional	contact your healthcare provider, talk to your doctor, seek medical attention, consult your doctor	**0.523**
PC	Snippet has reassurance	mild|common|temporary|normal|usually resolves|should resolve	**0.518**
PC	Snippet has causal hedging	(can|may) (cause|lead to|result in)	0.512
PC	Explanation supports Snippet	is consistent with	0.518
PC	Explanation contains omission phrases	((important|crucial|essential|fails) to|does not|doesn't|omits|lacks) (mention|state|note|emphasize|highlight)	0.489
PC	Explanation critiques risk framing	downplay, understate risk, minimize risk, risk framing	**0.523**
PC	Explanation references guidelines	guidelines, current evidence, literature, DSM, criteria	**0.526**
PC	Explanation overly long	Threshold at 55 words	0.479
PC	Explanation critiques framing	may mislead patients|create barriers|framing|overemphasizing	0.516

Table 5: Full list of rules, with corresponding effects on F1 score. Improved F1 scores are shown in **bold**.