EM_Mixers at MEDIQA-CORR 2024: Knowledge-Enhanced Few-Shot In-Context Learning for Medical Error Detection and Correction

Swati Rajwal[1], Eugene Agichtein[1], Abeed Sarker[2]

[1]Department of Computer Science & Informatics, Emory University
[2]Department of Biomedical Informatics, Emory University
Correspondence: srajwal@emory.edu

Abstract

This paper describes our submission to MEDIQA-CORR 2024 shared task for automatic identification and correction of medical errors in a given clinical text. We report results from two approaches: the first uses a few-shot in-context learning (ICL) with a Large Language Model (LLM) and the second approach extends the idea by using a knowledge-enhanced few-shot ICL approach. We used Azure OpenAI GPT-4 API as the LLM and Wikipedia as the external knowledge source. We report evaluation metrics (accuracy, ROUGE, BERTScore, BLEURT) across both approaches for validation and test datasets. Of the two approaches implemented,[1] our experimental results show that the knowledge-enhanced few-shot ICL approach with GPT-4 performed better with error flag (subtask A) and error sentence detection (subtask B) with accuracies of 68% and 64%, respectively on the test dataset. These results positioned us fourth in subtask A and second in subtask B, respectively in the shared task.

1 Introduction

An estimated 795,000 Americans either become permanently disabled or die each year across various healthcare settings due to misdiagnoses of serious diseases, as reported by Newman-Toker et al. (2024). The key process failures, especially in the emergency department, are errors in diagnostic assessment, test ordering, and test interpretation (Newman-Toker et al., 2023). Therefore there is a growing interest to assist clinicians in automatic medical error identification, if any, in a clinical note. The MEDIQA-CORR 2024 shared task (Ben Abacha et al., 2024a), hosted by the 6th Clinical Natural Language Processing Workshop at NAACL 2024, was proposed to encourage research in medical error identification and correction in clinical texts. From a human perspective, these errors require medical expertise and knowledge to be both identified and corrected. Here we describe our submission to the three sub-tasks: error detection, error sentence identification, and error correction. We explore two approaches; the first uses LLM for error detection and correction while the second extends the approach by integrating an additional layer of information retrieval. We selected GPT-4 since it has shown good performance on a variety of medical tasks, according to various recent studies (Nori et al., 2023; Waisberg et al., 2023; Gertz et al., 2024). Out of the two approaches discussed here and implemented, we observed that the second approach performed better as measured by the evaluation metrics (section 4). The results for error flag (sub-task A) and error sentence detection (sub-task B) by our proposed system (approach 2) ranked fourth and second, respectively, in the shared task.

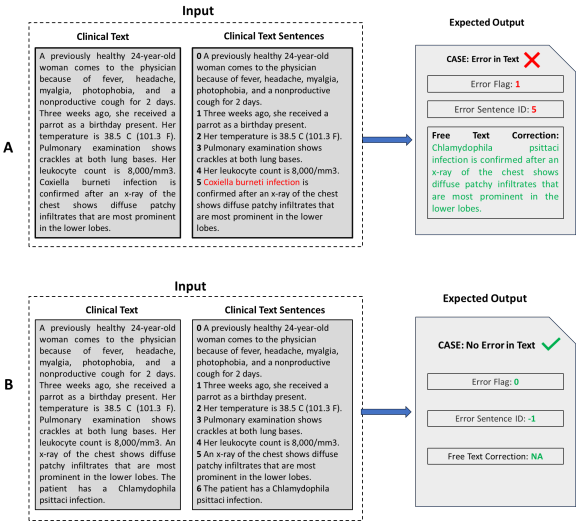


Figure 1: Example of clinical texts and clinical text sentences from the training set (Ben Abacha et al., 2024b) that have (A) a medical error and (B) no medical error.

[1]https://github.com/swati-rajwal/EM_Mixers_MEDIQA-CORR-NAACL-ClinicalNLP-2024 (last accessed: 04/24/2024)

Proceedings of the 6th Clinical Natural Language Processing Workshop, pages 590–595

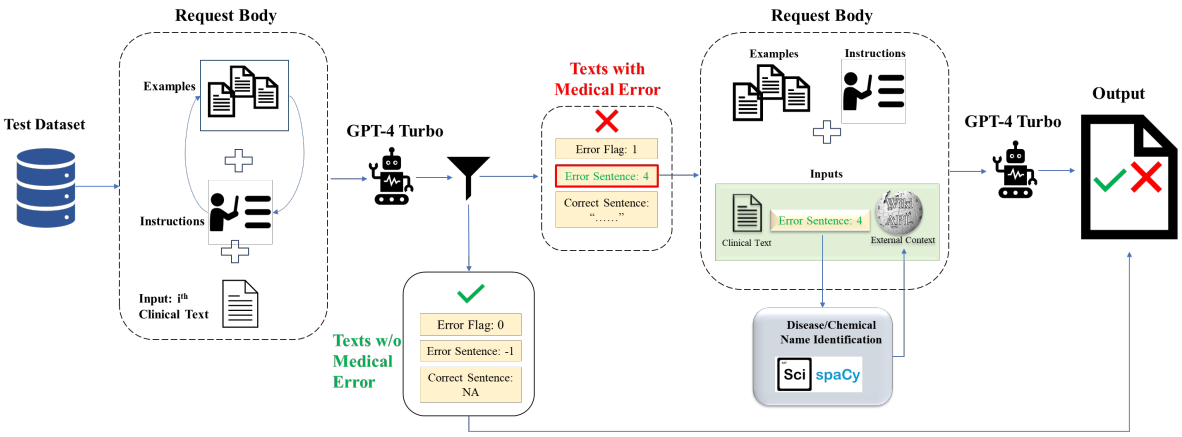


Figure 2: Outline of the proposed approach, illustrating the LLM and information retrieval components.

2 Shared Task and Dataset

MEDIQA-CORR 2024 proposed the following three sub-tasks. Each sub-task builds upon the previous one, creating a sequential process for detecting, identifying, and correcting errors in medical texts.

1. **Sub-Task A (Medical Error Identification/Binary Classification)**: Given a patient's clinical text, the task is to detect whether the text includes a medical error.

2. **Sub-Task B (Erroneous Sentence/Span Identification)**: If an error is identified in the given clinical text, the next task is to identify the text span associated with the error if a medical error exists.

3. **Sub-Task C (Correction of Erroneous Sentence)**: If the given clinical text has a medical error, this task requires rectifying or correcting the erroneous text span and providing a free text correction.

2.1 Dataset

The dataset (Ben Abacha et al., 2024b) was provided by two institutions: Microsoft (MS) and the University of Washington (UW). Specifically, the training dataset (MS) consists of 2,189 examples. The validation dataset contains 734 examples (574 from MS and 160 from UW, respectively) and the test set contains 925 samples. Each sample contains "Text ID" (unique), "Text" (clinical note), and "Sentences" (clinical note divided into sentences with IDs). Additionally, the training and validation dataset contains ground truth values under the columns: "Error Flag" (0 for no error, 1 otherwise), "Error Sentence ID", "Error Sentence", "Corrected Sentence", and "Corrected Text". The mean length of a clinical text in the training dataset is 781 words (Fig. 3). The clinical text contains critical information such as symptoms, clinical examination findings, patient history, and other details. Figure 1 shows the two possible cases in the dataset—either there is a medical error in the given clinical text or there is none.

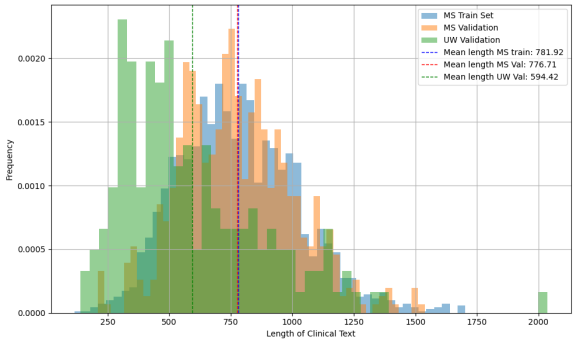


Figure 3: Clinical text lengths across datasets.

3 Proposed Approach

Figure 2 shows the entire framework and the following is the description of the two approaches to this shared task. We used GPT-4 as the LLM (Achiam et al., 2023) and designed a prompt to call the Microsoft Azure OpenAI GPT-4 Turbo (`gpt-4-1106-preview`) API[2]. This model has a context window of 128,000 tokens and returns a maximum of 4,096 output tokens. We set the `temperature` parameter to 0 and `top_p` to 0.95, respectively. For additional information and access

[2] https://platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo (last accessed: 04/24/2024)

to the code used in our study, please refer to the GitHub repository we have made publicly available.

3.1 Approach 1: Few-Shot In-Context Learning

For each clinical text, a request body for the GPT-4 model API is constructed as a set of instructions that outline the task of analyzing clinical text to identify and correct diagnostic errors. We provided 7 examples in the prompt to guide the LLM model in performing the analysis, followed by the actual clinical text and sentences to be evaluated. Fig. 4 shows the final prompt template which was curated over multiple manual iterations. Also, the examples in the prompts were taken from the training dataset only and remained constant across all the subsequent calls to the LLM API. The results for each clinical text were returned as a JSON object containing the error flag, erroneous sentence ID, and the corrected sentence (if any).

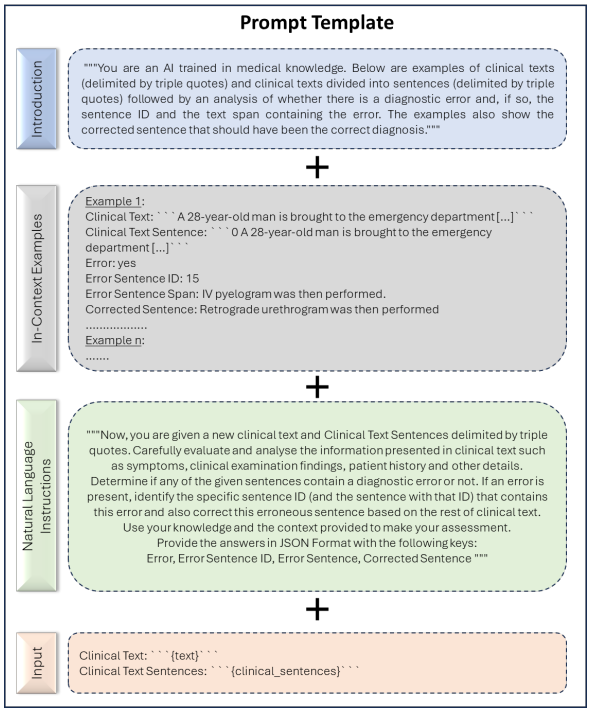


Figure 4: Prompt Template for the ICL-based approach.

3.2 Approach 2: Knowledge-Enhanced Few-Shot In-Context Learning

The first approach as described previously resulted in many positive predictions, especially false positives (i.e., predicted an error when there is none). Therefore, we decided to extend approach 1 (Fig. 2) by re-evaluating the instances that were previously identified by the GPT-4 model as positive (indicating the presence of an error). For such instances, an erroneous sentence was also predicted that forms a basis of re-evaluation in our approach 2.

To enrich the context for the GPT-4 model during its re-evaluation process, we integrated an additional layer of information retrieval. Specifically, we identified disease or chemical keywords within the sentence flagged as erroneous by using specialized models '`en_core_sci_scibert`' and '`en_ner_bc5cdr_md`' from Scispacy (Neumann et al., 2019). Then, we fetched related content from Wikipedia (English Wikimedia Wiki Endpoint[3]) and provided this external knowledge to GPT-4 alongside the original clinical text. The intention behind this strategy was to supply the model with a broader context to enable it to make more informed decisions regarding the presence of medical errors. Also, note that if there is no Wikipedia page for a particular keyword, then vanilla prompting is used (i.e., no context).

3.3 Rationale behind Scispacy models

ScispaCy is a Python package designed for processing biomedical, scientific, or clinical text using spaCy models. We utilized all eight available models to analyze the training dataset, specifically focusing on detecting keyword in the sentence marked as erroneous errors flagged by GPT-4. Our goal was to identify disease and chemical names in sentences where GPT-4 predicted errors. In our analysis, two models, '`en_core_sci_scibert`' and '`en_ner_bc5cdr_md`', worked well for keyword identification. Sometimes, `en_core_sci_scibert`' missed certain keywords that '`en_ner_bc5cdr_md`' could detect, and vice versa. Consequently, we decided to use both models to ensure comprehensive keyword detection. As an example, take a look at Figure A.1, which shows that most of the keywords of concern are detected by one or the other model.

3.4 Final Submission

Our final submission for the shared task included combined analysis through an ensemble method: For each instance if the error flag from Approach 1 is set to 0, the process moves to the next instance. If both approaches agree on the presence of an error (error flag = 1), the final result (dataframe in

[3] `https://en.wikipedia.org/w/api.php` (last accessed: 04/24/2024)

Table 1: Comparison of Approach 1 (few-shot in-context learning) and Approach 2 (knowledge-enhanced few-shot in-context learning) on validation and test datasets.

	Validation Dataset		Test Dataset	
Metric	Approach 1	Approach 2	Approach 1	Approach 2
Accuracy				
Error Flags Accuracy	0.622	0.648	0.626	**0.680**[a]
Error Sentence Detection Accuracy	0.598	0.638	0.562	**0.640**[b]
ROUGE Scores				
R1F_subset_check	0.488	0.550	0.540	0.571
R2F_subset_check	0.375	0.439	0.444	0.478
RLF_subset_check	0.481	0.543	0.534	0.565
R1FC	0.369	0.516	0.429	0.542
R2FC	0.313	0.484	0.388	0.512
RLFC	0.365	0.514	0.426	0.540
BERTScore				
BERTSCORE_subset_check	0.566	0.620	0.574	0.595
BERTC	0.407	0.537	0.444	0.550
BLEURT				
BLEURT_subset_check	0.569	0.607	0.580	0.596
BLEURTC	0.409	0.533	0.446	0.550
Average Composite Score				
aggregate_subset_check	0.541	0.592	0.565	0.587
AggregateC	0.395	0.529	0.440	0.548

[a] Fourth and [b] Second best accuracy in the shared task results among 17 participating teams.

Python) is updated with the error flag with the sentence ID from Approach 1, and the corrected sentence as identified. If Approach 1 flags an error but Approach 2 does not, the instance is left unchanged, moving on to the next. This methodical combination of inferences from both approaches forms our final solution for error identification and correction mechanism essentially giving more weightage to the knowledge-enhanced approach.

3.5 Evaluation

The official evaluation script[4] provided by the organizers was used for model evaluation. The test set results were released after system submission on codalab. The proposed systems predictions are evaluated for binary accuracy of error detection and a multi-dimensional evaluation of text correction quality against the provided ground truth notes with the following metrics: ROUGE (Lin, 2004), BERTScore (Zhang* et al., 2020), and BLEURT (Sellam et al., 2020).

[4] https://github.com/abachaa/MEDIQA-CORR-2024 (last accessed: 04/24/2024)

4 Results

Table 1 shows the results on the validation and test dataset for multiple evaluation metrics. Refer to Appendix A.1 for the detailed definition of each metric variable name.

Accuracy Metrics: Experimental results show that Approach 2 improved error flag accuracy by about 2.6% on the validation dataset and 5.4% on the test dataset. Similarly, for Error Sentence Detection Accuracy, Approach 2 shows an improvement of approximately 4% and 7.8% on the validation and test datasets, respectively. This suggests that providing external context around the disease/chemical name is useful (to a certain extent) for GPT-4 in making sound decisions.

ROUGE Scores: Approach 2 demonstrates a higher score compared to Approach 1, with improvements of approximately 6.2% and 3.2% on the validation and test datasets, respectively. Similar performance improvements were observed for BERTScore, BLEURT and Average Composite scores.

5 Discussion

Across multiple evaluation metrics and datasets, Approach 2 consistently outperforms Approach 1. This indicates that the addition of external knowledge is potentially leveraging more effective strategies for both error detection and error correction.

5.1 Error Analysis

We studied the misclassified examples in the dataset. It appears that the model found it challenging to recognize rare or complex conditions (e.g., Picornavirus, being less commonly referenced in lay texts). Although external information from Wikipedia is used to provide context, GPT-4's interpretation of this supplementary data is still limited by its ability to integrate and analyze it effectively within the clinical scenario presented. This process might have been complicated due to Wikipedia content being too general to aid in accurate analysis.

5.2 Limitations & Future Directions

Automatic evaluation metrics such as ROUGE, BERTScore, and BLEURT may not accurately reflect human judgment. Therefore, in real-life settings, it is necessary to conduct an expert human evaluation to validate the results. Furthermore, our current approach uses Wikipedia as the external source of information which, while a rich source of information, might not be very specialized for medical knowledge. In the future, we plan to utilize other sources of medical knowledge, such as PubMed. During the second approach, we rely solely on the sentence that has been predicted by GPT-4 to be erroneous. This might be wrong since there were cases when GPT-4 correctly identified that there was an error in the clinical text but incorrectly identified the erroneous sentence span which is the basis of our knowledge-retrieval component.

6 Conclusion

In this paper, we present our submission to the MEDIQA-Corr shared task for Medical Error Detection and Correction. We evaluated two approaches: one with in-context learning (ICL) and the other an extension with knowledge-enhanced few-shot ICL. Based on the evaluation metric results, we conclude that knowledge-enhanced few-shot in-context learning is a promising path toward medical error detection and correction. For future work, we plan to experiment the proposed pipeline with other sources of medical information for comparative analysis.

Acknowledgement

Thanks to Kaustubh Dhole and Harshita Sahijwani for the useful discussions around the task.

Data & Code Availability

https://github.com/swati-rajwal/EM_Mixers_MEDIQA-CORR-NAACL-ClinicalNLP-2024

Ethics Statement

Design and development of an automated system for medical error detection and correction can raise many ethical issues. For instance, the system design should address issues of data bias and fairness to avoid unfair medical error detection for certain patient groups. Also, transparency about the system's capabilities and limitations is key, allowing users to understand and trust our AI's decisions. We also emphasize the importance of sourcing credible information, particularly when integrating external content like Wikipedia, to maintain the accuracy and relevance of our corrections.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Asma Ben Abacha, Wen wai Yim, Velvin Fu, Zhaoyi Sun, Fei Xia, and Meliha Yetisgen. 2024a. Overview of the mediqa-corr 2024 shared task on medical error detection and correction. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, Mexico City, Mexico. Association for Computational Linguistics.

Asma Ben Abacha, Wen wai Yim, Velvin Fu, Zhaoyi Sun, Meliha Yetisgen, Fei Xia, and Thomas Lin. 2024b. Medec: A benchmark for medical error detection and correction in clinical notes. *CoRR*.

Roman Johannes Gertz, Thomas Dratsch, Alexander Christian Bunck, Simon Lennartz, Andra-Iza Iuga, Martin Gunnar Hellmich, Thorsten Persigehl, Lenhard Pennig, Carsten Herbert Gietzen, Philipp Fervers, David Maintz, Robert Hahnfeldt, Jonathan Kottlors, and Linda Moy. 2024. Potential of gpt-4 for detecting errors in radiology reports: Implications for reporting accuracy. *Radiology*, 311(1):e232714. PMID: 38625012.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy. Association for Computational Linguistics.

David E Newman-Toker, Najlla Nassery, Adam C Schaffer, Chihwen Winnie Yu-Moe, Gwendolyn D Clemens, Zheyu Wang, Yuxin Zhu, Ali S Saber Tehrani, Mehdi Fanai, Ahmed Hassoon, et al. 2024. Burden of serious harms from diagnostic error in the usa. *BMJ Quality & Safety*, 33(2):109–120.

David E Newman-Toker, Susan M Peterson, Shervin Badihian, Ahmed Hassoon, Najlla Nassery, Donna Parizadeh, Lisa M Wilson, Yuanxi Jia, Rodney Omron, Saraniya Tharmarajah, et al. 2023. Diagnostic errors in the emergency department: a systematic review.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Ethan Waisberg, Joshua Ong, Mouayad Masalkhi, Sharif Amit Kamran, Nasif Zaman, Prithul Sarker, Andrew G Lee, and Alireza Tavakkoli. 2023. Gpt-4: a new era of artificial intelligence in medicine. *Irish Journal of Medical Science (1971-)*, 192(6):3197–3200.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix

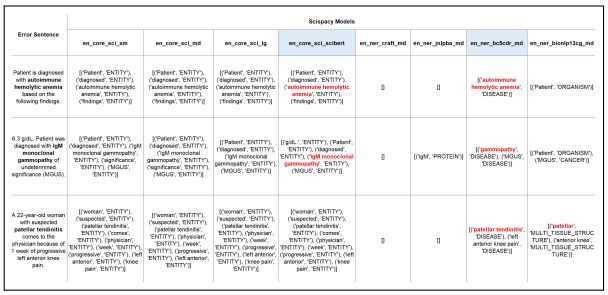
A.1 Evaluation Metrics

‘aggregate_subset_check’ is the mean score of all individual metric scores combined for each subset of data.

‘R1F_subset_check’ is the F_1-score of the ROUGE-1 metric and assesses how many of the same words are used in both texts, adjusted for both precision and recall.

‘R2F_subset_check’ is the F_1-score of the ROUGE-2 metric, focusing on the overlap of bigrams between the generated and reference texts.

Figure A.1: ScispaCy models for entity detection.



Error Sentence	Scispacy Models							
	en_core_sci_sm	en_core_sci_md	en_core_sci_lg	en_core_sci_scibert	en_ner_craft_md	en_ner_jnlpba_md	en_ner_bc5cdr_md	en_ner_bionlp13cg_md
Patient is diagnosed with **autoimmune hemolytic anemia** based on the following findings.	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('autoimmune hemolytic anemia', 'ENTITY'), ('findings', 'ENTITY')]	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('autoimmune hemolytic anemia', 'ENTITY'), ('findings', 'ENTITY')]	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('autoimmune hemolytic anemia', 'ENTITY'), ('findings', 'ENTITY')]	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), 'autoimmune hemolytic anemia', 'ENTITY'), ('findings', 'ENTITY')]	[]	[]	[('autoimmune hemolytic anemia', 'DISEASE')]	[('Patient', 'ORGANISM')]
6.3 g/dL. Patient was diagnosed with **IgM monoclonal gammopathy** of undetermined significance (MGUS).	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('IgM monoclonal gammopathy', 'ENTITY'), ('significance', 'ENTITY'), ('MGUS', 'ENTITY')]	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('IgM monoclonal gammopathy', 'ENTITY'), ('significance', 'ENTITY'), ('MGUS', 'ENTITY')]	[('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), ('IgM monoclonal gammopathy', 'ENTITY'), ('MGUS', 'ENTITY')]	[('g/dL.', 'ENTITY'), ('Patient', 'ENTITY'), ('diagnosed', 'ENTITY'), 'IgM monoclonal gammopathy', 'ENTITY'), ('MGUS', 'ENTITY')]	[]	[('IgM', 'PROTEIN')]	[('gammopathy', 'DISEASE'), ('MGUS', 'DISEASE')]	[('Patient', 'ORGANISM'), ('MGUS', 'CANCER')]
A 22-year-old woman with suspected **patellar tendinitis** comes to the physician because of 1 week of progressive left anterior knee pain.	[('woman', 'ENTITY'), ('suspected', 'ENTITY'), ('patellar tendinitis', 'ENTITY'), ('comes', 'ENTITY'), ('physician', 'ENTITY'), ('week', 'ENTITY'), ('progressive', 'ENTITY'), ('left anterior', 'ENTITY'), ('knee pain', 'ENTITY')]	[('woman', 'ENTITY'), ('suspected', 'ENTITY'), ('patellar tendinitis', 'ENTITY'), ('physician', 'ENTITY'), ('week', 'ENTITY'), ('progressive', 'ENTITY'), ('left anterior', 'ENTITY')]	[('woman', 'ENTITY'), ('suspected', 'ENTITY'), ('patellar tendinitis', 'ENTITY'), ('physician', 'ENTITY'), ('week', 'ENTITY'), ('progressive', 'ENTITY'), ('left anterior', 'ENTITY'), ('knee pain', 'ENTITY')]	[('woman', 'ENTITY'), ('suspected', 'ENTITY'), ('patellar tendinitis', 'ENTITY'), ('physician', 'ENTITY'), ('week', 'ENTITY'), ('progressive', 'ENTITY'), ('left anterior', 'ENTITY'), ('knee pain', 'ENTITY')]	[]	[]	[('patellar tendinitis', 'DISEASE'), ('left anterior knee pain', 'DISEASE')]	[('patellar tendinitis', 'MULTI_TISSUE_STRUCTURE'), ('anterior knee', 'MULTI_TISSUE_STRUCTURE')]

‘RLF_subset_check’ is the score for the ROUGE-L metric and measures the longest common subsequence between the generated and reference texts.

‘R1FC’, ‘R2FC’, and ‘RLFC’ are composite scores for the ROUGE-1, ROUGE-2, and ROUGE-L metrics, respectively, adjusted for the total number of texts, including those correctly identified as no error ("NA" cases). These scores balance between correctly generated corrections and correctly identified non-correction scenarios.

‘BERTSCORE_subset_check’ reflects the mean BERTScore F_1 metric and uses BERT’s contextual embeddings to compare the generated text against references. ‘BERTC’ is the composite score for BERTScore, taking into account the entire dataset and adjusting for "NA" cases similar to the ROUGE composite scores.

‘BLEURT_subset_check’ represents the mean BLEURT score for the subsets of data. BLEURT is a learned metric that compares generated text to reference texts, fine-tuned on human judgments.

‘BLEURTC’ is the composite score for BLEURT, adjusted for the total dataset including "NA" scenarios.

‘AggregateC’ is the average composite score of all individual metrics (ROUGE-1 F_1, BERTSCORE, BLEURT), providing a single, consolidated measure of the NLG system’s performance across the entire evaluation framework.