LAB-KG: A Retrieval-Augmented Generation Method with Knowledge Graphs for Medical Lab Test Interpretation

Rui Guo[1,2], Barry Devereux[2], Greg Farnan[1], Niall McLaughlin[2],
[1]Cirdan, [2]Queen's University Belfast
Correspondence: rui.guo@cirdan.com

Abstract

Laboratory tests generate structured numerical data, which a clinician must interpret to justify diagnoses and help patients understand the outcomes of the tests. LLMs have the potential to assist with the generation of interpretative comments, but legitimate concerns remain about the accuracy and reliability of the generation process. This work introduces LAB-KG, which conditions the generation process of an LLM on information retrieved from a knowledge graph of relevant patient conditions and lab test results. This helps to ground the text-generation process in accurate medical knowledge and enables generated text to be traced back to the knowledge graph. Given a dataset of laboratory test results and associated interpretive comments, we show how an LLM can build a KG of the relationships between laboratory test results, reference ranges, patient conditions and demographic information. We further show that the interpretive comments produced by an LLM conditioned on information retrieved from the KG are of higher quality than those from a standard RAG method. Finally, we show how our KG approach can improve the interpretability of the LLM generated text.

1 Introduction

Artificial Intelligence (AI) has become increasingly influential in the medical field, offering transformative potential in various applications such as medical data summarisation (Van Veen et al., 2024) and diagnostics (Tu et al., 2024). The data generated in clinical care, from Electronic Health Records (EHRs) to laboratory tests, present both an opportunity and a challenge. In principle, using such data efficiently and intelligently has the potential to create efficiencies for healthcare professionals which allow them to improve patient experiences and outcomes. Laboratory diagnostics generate substantial amounts of structured numerical data, which can be difficult for patients and clinicians to interpret effectively. AI models have the potential to provide interpretative comments and personalised explanations of laboratory results, improving the laboratory-clinical interface, and improving patient understanding (Padoan and Plebani, 2022a,b).

However, there are critical considerations when using AI models in the medical domain, including issues such as hallucinations, inaccuracies, and non-determinism. These issues can lead to incorrect or harmful results in healthcare (Cadamuro et al., 2023; Stevenson et al., 2024), and the errors can often be difficult to identify during model evaluation and to characterize *a priori*. These problems call for approaches to improve the reliability and accuracy of AI systems in medicine.

Integrating Knowledge Graphs (KGs) with LLMs through Retrieval-Augmented Generation (RAG) can be a promising strategy. KGs provide structured, interconnected data that can ground LLM outputs in factual information, reducing hallucinations, and improving the accuracy of AI-generated content (Yan et al., 2024; Gilbert et al., 2024). By combining the LLM's generative capabilities with the KG's factual grounding, AI systems can be more reliable and explainable.

In this work, we aim to improve laboratory test interpretation generation by combining RAG with a Knowledge Graph, referred to as the LAB-KG approach. Traditional RAG methods rely on embedding similarity between the user's query and a set of documents or knowledge base entries. They retrieve relevant information to condition the language model's generation process. However, the reasoning behind the generated interpretations often remains a black box, Our LAB-KG approach uses both the internal knowledge of LLMs and lab test examples to build a knowledge graph that explicitly captures the relevance between each test result and the patient's condition. This allows for more explainable and transparent interpretation generation.

Bridging Neurons and Symbols for Natural Language Processing and Knowledge Graphs Reasoning @ COLING 2025, pages 40–50
January 20th, 2025.

Our contributions are threefold:

1. **Knowledge Graph Construction with Limited Examples**: We present a novel approach for building a Knowledge Graph (KG) utilising the internal knowledge of Large Language Models (LLMs) and a limited set of laboratory test examples, capturing the relationships between test results and medical conditions.

2. **Improved Performance over Retrieval-Augmented Generation (RAG)**: Our KG-based approach demonstrates better performance compared to traditional Retrieval-Augmented Generation methods. By structurally representing knowledge, the system can more accurately interpret and retrieve relevant conditions from new patient test results.

3. **Explainable System**: The proposed KG approach offers greater interpretability than standard RAG methods. The explicit structure of the KG allows for the tracing of errors in generated reports back to specific nodes and relationships within the graph.

2 Previous Work

The application of AI to the task of laboratory test interpretation is an area of growing interest. Traditional methods of providing interpretative comments on laboratory reports have been recognised as essential to improving the laboratory-clinical interface (Plebani, 2009).

Several studies have been applied to use AI and natural language processing models to interpret laboratory test results. Cadamuro et al. (2023) evaluated the performance of ChatGPT and other AI models in understanding laboratory medicine test results. Whilst the AI models could recognise laboratory tests and detect deviations from reference intervals, their interpretations were often superficial and incorrect. The models sometimes failed to differentiate between slight and severe deviations and did not provide meaningful suggestions for follow-up diagnostics.

Stevenson et al. (2024) evaluated the thyroid function test result interpretation by biochemist, ChatGPT, and Google Bard. The AI tools correctly interpreted only a fraction of the cases, showing the limitations of current AI models in complex medical interpretation tasks.

Abusoglu et al. (2024) assessed the performance of various chatbots as assistants for problem-solving in clinical laboratories. Their study showed that AI applications had good performance in identifying cases and responding to questions related to preanalytical, analytical, and postanalytical errors. However, the chatbots' accuracy varied, and there were concerns about their reliability and safety in clinical settings.

An early work by Patil et al. (2013) introduced a Concept Graph Engine (CG-Engine) that generates patient-specific personalised disease rankings based on laboratory test data, using the Unified Medical Language System (UMLS) as a medical knowledge base. The CG-Engine constructs a concept graph connecting laboratory tests to diseases and computes weights based on relation types, semantic types, and other attributes. While their approach utilises a knowledge base to connect lab tests and conditions, it relies on pre-existing medical ontologies that may differ from the actual data terminology.

Despite these advancements, a major challenge with LLMs in the medical domain is their tendency to produce hallucinations and inaccurate information. Retrieval-Augmented Generation (RAG) techniques have been proposed to mitigate these issues, where LLMs are augmented with external knowledge sources to ground their outputs in factual data. Zakka et al. (2024) developed *Almanac*, an LLM framework augmented with retrieval capabilities from curated medical resources for medical guidelines and treatment recommendations. Their results showed significant performance improvements compared to standard LLM pipelines.

In the domain of laboratory test interpretation, He et al. (2023) built a dataset by collecting and annotating interpretations of textual lab results from health articles. They evaluated transformer-based language models for recognizing key terms and mapped them to concepts in major controlled terminologies.

In healthcare generally, integrating LLMs with Knowledge Graphs can improve the reliability and accuracy of AI models. Gilbert et al. (2024) discussed the potential of combining LLMs with KGs as medical information curators. By providing a structured representation of medical knowledge, KGs can help LLMs generate more accurate and verifiable outputs, reducing the risk of misinformation and enhancing patient safety.

Our work builds upon these approaches by constructing a knowledge graph that combines LLM internal knowledge with examples to associate lab

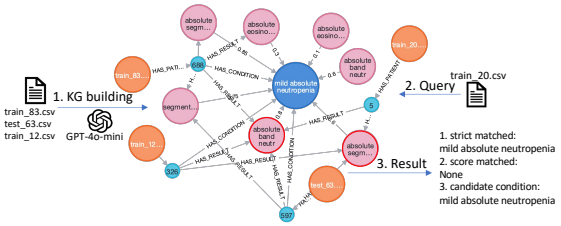

Figure 1: Overview of LAB-KG. GPT-4o-mini helps to find the relationship between the test result and condition. When a new patient's test results are input into the system, they are compared with the LAB-KG to identify relevant conditions using strict set matching and confidence score matching. This process enables the generation of accurate and explainable interpretations of laboratory test results.

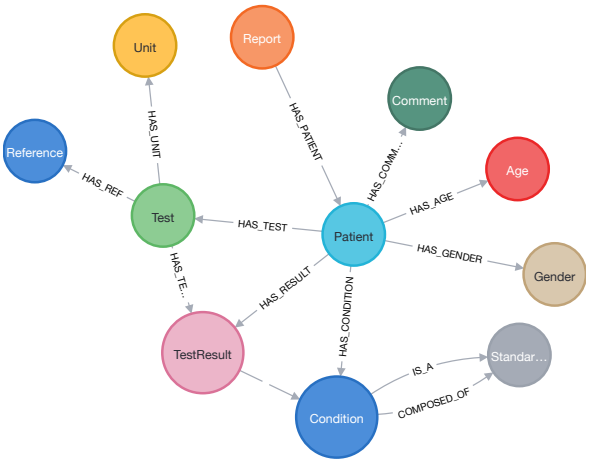

Figure 2: The schema of the lab-kg.

tests with conditions. This allows for improved accuracy in lab test interpretation generation and provides explainability through the graph structure.

3 Method

Given a set of patient full blood test csv files, we build a LAB-KG with the help of GPT-4o-mini to find the relationships between the test results and conditions. A condition in our context refers to a specific medical finding or diagnosis derived from laboratory test results. For instance, "Mild normochromic normocytic anaemia" indicates a type of anemia characterized by red blood cells that are of normal size (normocytic) and normal hemoglobin content (normochromic). Clinicians use those conditions to determine the appropriate follow-up and management for the patient. The new patient test result is compared with the LAB-KG to find the relevant condition. An overview of this process is in Figure 1. An example of a transcribed report is shown in Table 1.

3.1 KG-RAG Approach

We propose an approach combining both the internal knowledge of a large language model and limited examples to build a knowledge graph. A laboratory test is a medical procedure using a sample of blood, urine, or other tissues to assess a patient's health. Interpreting lab test results can be complex due to the subtle variations that may indicate different medical conditions. Our knowledge graph (KG) represents relationships between lab tests, conditions, and patients and can be queried to generate interpretations for new patients. The relation between lab tests and conditions are built as in the method described below.

3.1.1 Graph Building

The knowledge graph is constructed to model the relationships between lab tests, patients, conditions, and results. The schema of the graph is shown in Figure 2. The key nodes and relationships are summarised in Table 4 in appendix.

Reference Ranges A reference range is the set of values considered normal for a healthy individual for a specific test, serving as a benchmark to interpret individual test results. The reference ranges for some tests are sometimes missing, and to address this issue, we aggregated all the reports' reference. We ask LLM to infer the correct reference range by providing all the related reference ranges for that test and asking the LLM to use its internal medical knowledge for the patient.

Test Names and Test Result Test names are the standardised identifiers used to represent specific laboratory tests. The test names in our dataset are standardised by curating a list of test names and manually mapping different variations to a standard name. Each *TestResult* node represents the result of a specific test. If the reference range is provided, the test result will be labelled with a suffix indicating its status (e.g., *Normal*, *Abnormal (High)*, *Abnormal (Low)*, *Borderline (High)*, *Borderline (Low)*).

Condition Extraction The most important task for LLM is extracting the conditions from the comments and determining the relevance of each test result to the conditions mentioned in the patient comments. We prompted the LLM to split the comment into several conditions and establish potential *CONTRIBUTES_TO* relationships between each test result node and condition node. This effectively

Category	Test Name	Result	Unit	Ref Start	Ref End	norm	Ab flag
Info	Age	9					
Blood	Haemoglobin	11.30	g/dL	11.5	15.5	-0.05	Low
Blood	Hematocrit	33.9	%	35	45	-0.11	Low
Blood	Red cell count	4.71	x10^6/uL	4	5.2	0.59	
Blood	MCV	72.0	fL	78	96	-0.33	Low
Blood	MCH	24.0	pg	26	32	-0.33	Low
Blood	MCHC	33.4	g/dL	31	36	0.48	
Blood	RDW	14.2	%	11.5	14.5	0.9	
Blood	Platelet Count	292	x10^3/uL	170	450	0.44	
Blood	T.L.C	8.2	x10^3/uL	5	13	0.4	
WBC Diff	Basophils	1	%	0			
...	...						
WBC Diff	Monocytes (Absolute)	1.1	x10^3/uL	0.2	1	1.12	High
Comments	Mild microcytic hypochromic anaemia. Platelets are adequate. Mild absolute monocytosis.						

Table 1: Patient report example (Abridged). Ab flag: abnormality flag.

builds a rule set based on the examples and the LLM's knowledge. For example, "Mild normocytic normochromic anemia with mild anisocytosis" can be split into two conditions: "Mild normocytic normochromic anemia" and "mild anisocytosis." We only ask LLM to infer that *CONTRIBUTES_TO* relationship from the abnormal conditions to test results, and omit the conditions such as "normal blood picture" or "follow up is recommended", which cannot be mapped to a set specific test result.

Knowledge Aggregation We added an aggregation stage where we asked the LLM to assign weights to each relationship between a test result and a condition identified by the LLM. First, we added a *StandardTerm* node to group different conditions with potential semantic similarity. This grouping is based on querying each condition name using the BioPortal API for standardised terms, prioritizing matches in ontologies such as SNOMED CT, LOINC, and MEDDRA. In this way, we can group conditions under the same standard term, such as "mild anaemia" and "moderate anaemia" both being under the standard term "anaemia." Then, by providing all the *CONTRIBUTES_TO* in the KG between a condition group and related *TestResult*, we aim for the LLM to use these examples to indicate the importance of each test result for a particular condition group by assigning a weight to each *CONTRIBUTES_TO* relationship. This weight-assigning stage uses the aggregation *CONTRIBUTES_TO* from a condition with the frequency of each test result and the patient age/gender distribution.

3.2 Graph Retrieval Process

The KG is queried to find candidate conditions for a new patient. We tested three methods to find relevant conditions: an example-based match, a confidence score ranking, and their combination.

We first identify abnormal test results for a new patient and retrieve the connected *Condition* nodes for any abnormal tests in that patient, creating a list of candidate conditions. The connected patients and their related test results for each potential condition are retrieved from there.

Not all test results connected to a condition are critical; some might be false positives or less relevant. To filter less important conditions, we use two methods to select potential conditions.

Strict Match For each condition, we compare the test results of the new patient to those of the retrieved patients. Suppose the test results of the new patient cover all the test results of one patient in the training dataset connected to that condition (here, *Borderline* and *Abnormal* are treated the same). We consider it a "strict match" for that condition. An example is illustrated in Figure 3, with the condition "mild normochromic normocytic anemia." The new patient (with id 938) matches most of the test results of an existing patient (with id 100) but lacks "RBC count Abnormal (Low)." In this case, the new patient will not be assigned to this condition based on strict match. Note that there are other test results related to the condition without

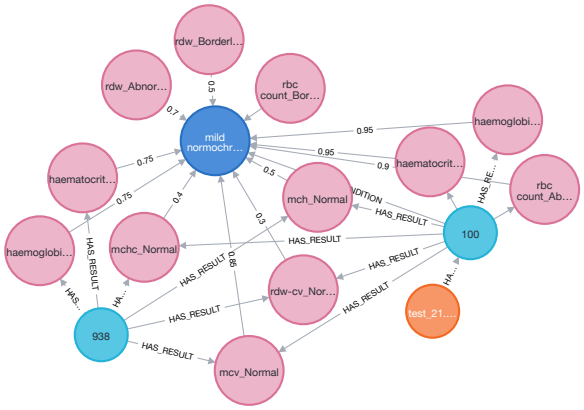

Figure 3: The condition query process for a new patient. The patient with ID 938 lacks "RBC count Abnormal (Low)" compared to the example patient with ID 100 using strict test result matching.

Test result	Weight
haematocrit_Abnormal (Low)	0.95
haemoglobin_Abnormal (Low)	0.95
rbc count_Abnormal (Low)	0.9
mcv_Normal	0.85
haemoglobin_Borderline (Low)	0.75
haematocrit_Borderline (Low)	0.75
rdw_Abnormal (High)	0.7
rbc count_Borderline (Low)	0.6
rdw_Borderline (High)	0.5
mch_Normal	0.5
mchc_Normal	0.4
rdw-cv_Normal	0.3

Table 2: The weight for "mild normochromic normocytic anemia" assigned by LLM

any patient connected, due to additional borderline connections added with slightly lower weights than abnormal, or because there are existing patients in the same condition group with those test results.

Confidence score-based match We utilise the weight assigned on the *CONTRIBUTES_TO* relationship to calculate each condition's confidence score by normalizing the weights connected to that patient for each condition. We sum the weights of the test results in the patient connected to one condition, and divide that sum by the total weight of all test results linked to that condition. A detailed example of a confidence score match is in the Appendix. The threshold to filter the confidence score is decided by the performance of the training data, as explained in section 4.

After the candidate conditions were retrieved from graph retrieval, we added an optional finalising stage using LLM to refine the conditions given the candidate conditions, merging potential duplicates or selecting the most specific condition rather than a broader one.

A key advantage of our LAB-KG approach is its inherent explainability addressing the limitations of traditional AI models in laboratory test interpretation. When generating interpretations for a new patient, clinicians can examine the specific test results leading to each suggested condition, along with the associated weights and confidence scores. This allows the clinicians to understand which conditions are being suggested and the rationale behind them. For instance, if a condition is identified, clinicians can review the exact match of test results between the new patient and existing examples and the weights of individual test results contribute to the overall confidence score.

Explainability Example The Knowledge Graph (KG) provides a transparent means to explain why each condition is retrieved, allowing us to identify and correct errors by examining the relationships between conditions and test results. As an illustrative example, consider the case of a patient diagnosed with "mild microcytosis," depicted in Figure 4. Initially, the KG connected both low Mean Corpuscular Hemoglobin (MCH) and low Mean Corpuscular Volume (MCV) to "mild microcytosis," even though low MCV alone is sufficient to diagnose microcytosis. When querying a new patient (ID 283) who exhibited low MCV but not low MCH, the system failed to retrieve "mild microcytosis" because the KG's connections implied that both low MCH and low MCV were required for retrieval. Upon reviewing the definition of "mild microcytosis", we corrected the KG by removing the redundant connection between low MCH and "mild microcytosis." After this, the system successfully retrieved "mild microcytosis" for the patient, demonstrating how the explainability provided by the KG facilitates refinement and improves retrieval accuracy.

4 Implementation and Evaluation

To the best of our knowledge, there are very few publicly accessible datasets providing detailed laboratory test reports along with associated clinical interpretations. We utilised a dataset from Mendeley Data (Abdelmaksoud et al., 2022), which includes 260 clinical laboratory test reports issued by

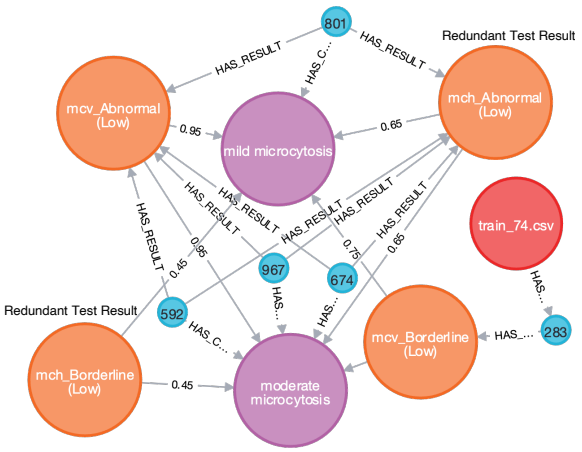

Figure 4: Illustration of explainability in the KG. Initially, "mild microcytosis" was connected to both "low MCH" and "low MCV." When querying patient ID 283, "mild microcytosis" was not retrieved because the KG incorrectly required both "low MCH" and "low MCV" for retrieval. After removing the unnecessary connection to "low MCH," "mild microcytosis" was successfully retrieved for the patient.

24 Egypt laboratories covering several test types. Among these, blood tests constitute the majority. We used GPT-4o [1] to transcribe all the blood test reports from PDF to CSV format. After removing duplicates, we obtained 47 unique blood test reports.

We used the Cypher query language in Python to build the KG in Neo4j Community Edition. GPT-4o-mini was used as the default LLM. Once the graph was built on training examples, it included 38 patient examples, 37 conditions, 78 test results, 459 nodes and 2287 relationships.

Our evaluation metrics include MEDCON (Yim et al., 2023), BLEU-3 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang et al., 2019), METEOR (Banerjee and Lavie, 2005), recall, precision, and F1 score. The LLM helps to preprocess the condition result, aligning semantically equivalent conditions (e.g., "mild anemia" vs. "anemia") between the generated and target, so that the extracted sets of conditions are comparable. The calculation of recall and precision itself remains a standard statistical comparison after the conditions have been extracted and aligned by the LLM. The F1 score is based on recall and precision. We use BLEU-3 instead of BLEU-4 because the results can be very short, such as "normal blood picture," which BLEU-4 would omit.

[1] https://platform.openai.com/docs/models

MEDCON is a metric for evaluating medical condition extraction from generated texts, considering semantic similarity and clinical relevance. The KG and KG with *CONTRIBUTES_TO* relationships inferred without examples (referred to KG * below) are compared for those metrics, together with other methods listed below.

In all our experiments, we performed five-fold cross-validation, with test data sizes of 10, 10, 9, 9, and 9 in each fold. We use MEDCON to select the threshold for the confidence score in each fold and from a list of values ranging from 0.1 to 0.9, with step 0.05. The best threshold values are stable across folds (0.55, 0.6, 0.55, 0.5, 0.5 for KG and 0.45, 0.35, 0.35, 0.35, 0.35 for KG *). The median values 0.55 and 0.35 are selected as the final threshold values for all the folds in KG and KG * respectively.

We compared the performance of the LAB-KG approach with several baseline methods, including:

1. Prompt Engineering

   A detailed prompt was designed to output different conditions given the patient report, which is a textual representation of each CSV file.

2. Text Embedding-Based Retrieval

   This method relies purely on text embeddings to retrieve relevant interpretations. The eight most similar examples are provided to the LLM for few-shot learning (we selected eight by testing numbers from 1 to 8). The query is the document of the new patient without the comment row. HuggingFace's `all-MiniLM-L6-v2` model embeds the text. We tested different document components in the retrieval and generation stages, including:

   - (A) Using all the rows.
   - (B) Using only the abnormal rows.
   - (C) Adding the normalised value as a column.

   An exhaustive search of all possible component combinations in the retrieval/generation stages is infeasible, so we tested four configurations using the same components for both retrieval and generation stages, and two configurations using different elements, totalling six results, as described in Table 3. The input examples include the above components and

the final comment, which the LLM may use as context to align its knowledge to the format and content of the example output comment.

3. LAB-KG Built with Examples (KG)

   We evaluated the results of the strict match, the confidence score match, and the combination of the strict and confidence score match. We also tested the effect of using the LLM to finalise the result.

4. Finding the relationship between *TestResult* and *Condition* without examples (KG *)

   An approach using the LLM's internal knowledge only to infer the *CONTRIBUTES_TO* relationships between test result nodes and condition nodes. We aim to assess the LLM's ability to find these relationships without examples. Based on the KG built with examples, the *CONTRIBUTES_TO* relationships are removed first. Then, for each condition, the LLM is provided with all possible test results to find the relationships and assign weights based on its own knowledge. New test result nodes can be created in this case.

5. Random Forest

   This traditional machine learning classifier was trained to predict the conditions given the patient data. Two kinds of inputs are tested: one with test results categorised as inputs (e.g., *Haemoglobin Abnormal (High)*), and another using the numerical test values directly. The conditions are classified, and adjectives such as "mild" and "moderate" are removed to reduce the possible classes to predict.

To determine whether using the LLM to evaluate the results is reliable, the correlation between F1 and each metric is shown in Table 5 in appendix. The F1 score has the highest correlation with MEDCON (0.95) and Bert score (0.94), and the correlation for the KG without examples is MEDCON (0.97) and Bert score (0.91). This validates the LLM's alignment between the generated and target results.

The results are presented in Table 3. The results show that combining the LLM's internal knowledge and examples can most effectively utilise the LLM and data, with an F1 score of 0.76, higher than the best KG * result of 0.71. The RAG approach has a best F1 score of 0.56, much lower than the best KG retrieval approach. When using the strict match, because it is based on the occurrence of test results in the examples, KG AND KG * show little difference. The combination of result of strict match and confidence score based match achieved higher score than separate result for KG, however, the combination of result for KG * is worse. The finalisation step does not make the result much different for the F1 score.

A detailed example about the difference in the result using KG and KG * in the appendix. All the LLM generated interpretation and the calculated metrics can be downloaded at https://docs.google.com/spreadsheets/d/10YTnKbLUs9UAJVGACh3wcNt-erMBLpJI

5 Conclusion

In this paper, we integrate the knowledge graph with RAG and LLM to improve the interpretation of laboratory test results with limited examples, providing an explainable framework clinicians can understand.

The evaluation demonstrated that the LAB-KG method outperforms LLM prompt engineering, text embedding-based retrieval, and random forests. The combination of strict matching and confidence score-based matching with KG allows us to retrieve the most clinically relevant interpretations. The KG with the relationship between condition and test result built without examples also performs well, especially in the strict match case, demonstrating its accurate internal knowledge.

We observed that in some cases, the relevance inferred by the LLM without examples was better than when examples were provided. This suggests the potential to combine the LLM's internal knowledge more effectively with examples to optimise performance. When multiple conditions are present in the example, the LLM sometimes struggles to differentiate the test results associated with each condition. Providing separate conditions in examples or generating synthetic data could help mitigate this issue.

The strength of our findings may be limited with only 47 blood test reports. Expanding the dataset and applying LAB-KG to other laboratory tests are essential steps for validating LAB-KG. The LLM's internal knowledge may not be up to date and limited, and integrating our knowledge graph with external medical ontologies like SNOMED CT is for future exploration.

category		medcon	bleu	bert score	meteor	rouge1	rouge2	rougeL	recall	precision	f1
zero shot	PE only	0.28	0.16	0.5	0.25	0.31	0.15	0.3	0.45	0.44	0.38
RAG	Q=G=A	0.39	0.29	0.6	0.44	0.47	0.32	0.45	0.56	0.46	0.48
	Q=G=A,C	0.38	0.2	0.59	0.39	0.42	0.23	0.39	0.56	0.42	0.46
	Q=G=B	0.49	0.29	0.64	0.47	0.5	0.32	0.48	0.68	0.5	0.56
	Q=G=B,C	0.47	0.28	0.63	0.46	0.48	0.3	0.47	0.67	0.51	0.56
	Q=A,C G=B,C	0.46	0.29	0.64	0.45	0.49	0.3	0.47	0.67	0.52	0.56
	Q=B G=B,C	0.48	0.29	0.64	0.47	0.49	0.31	0.48	0.67	0.51	0.56
KG retrieval	strict	0.68	0.37	0.71	0.56	0.56	0.43	0.53	0.78	0.65	0.67
	score	0.67	0.29	0.67	0.5	0.51	0.39	0.45	0.73	0.65	0.66
	strict + score	**0.75**	0.36	0.73	**0.58**	0.58	**0.47**	0.53	**0.88**	0.73	**0.76**
KG * retrieval	strict	0.67	0.3	0.7	0.51	0.54	0.38	0.5	0.78	0.65	0.68
	score	0.58	0.23	0.63	0.45	0.45	0.35	0.41	0.74	0.58	0.62
	strict + score	0.6	0.25	0.65	0.49	0.47	0.36	0.44	0.81	0.59	0.66
KG retrieval + finalise	strict	0.64	**0.41**	0.73	0.56	0.61	0.46	0.56	0.73	0.7	0.67
	score	0.59	0.28	0.67	0.49	0.53	0.35	0.46	0.69	0.68	0.65
	strict + score	0.74	0.4	**0.78**	0.58	**0.68**	0.45	**0.56**	0.82	**0.75**	0.75
KG * retrieval + finalise	strict	0.66	0.4	0.74	0.56	0.63	0.42	0.57	0.74	0.73	0.71
	score	0.51	0.27	0.66	0.46	0.53	0.34	0.47	0.68	0.55	0.57
	strict + score	0.55	0.3	0.68	0.51	0.54	0.36	0.46	0.7	0.64	0.65
random forest	input = categories				N/A				0.46	0.53	0.49
	input = values				N/A				0.28	0.38	0.32

Table 3: Evaluation results for different methods. PE: prompt engineering. RAG: A = using all the rows; B = using abnormal rows; C = adding normalised value as a column. The KG achieves the best result with strict match + confidence score, using the KG built with examples, with a F1 score 0.76. The KG * has a similar performance for the strict match, but worse with the confidence score and combination of strict match and confidence score. The best result for RAG is an F1 score of 0.56, which is higher than the zero-shot and random forest results.

References

Esraa Abdelmaksoud, Ahmed Gadallah, and Ahmed Asad. 2022. Clinical laboratory test reports.

Sedat Abusoglu, Muhittin Serdar, Ali Unlu, and Gulsum Abusoglu. 2024. Comparison of three chatbots as an assistant for problem-solving in clinical laboratory. *Clinical Chemistry and Laboratory Medicine (CCLM)*, 62(7):1362–1366.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Janne Cadamuro, Federico Cabitza, Zeljko Debeljak, Sander De Bruyne, Glynis Frans, Salomon Martin Perez, Habib Ozdemir, Alexander Tolios, Anna Carobene, and Andrea Padoan. 2023. Potentials and pitfalls of chatgpt and natural-language artificial intelligence models for the understanding of laboratory medicine test results. an assessment by the european federation of clinical chemistry and laboratory medicine (eflm) working group on artificial intelligence (wg-ai). *Clinical Chemistry and Laboratory Medicine (CCLM)*, 61(7):1158–1166.

Stephen Gilbert, Jakob Nikolas Kather, and Aidan Hogan. 2024. Augmented non-hallucinating large language models as medical information curators. *NPJ Digital Medicine*, 7(1):100.

Zhe He, Shubo Tian, Arslan Erdengasileng, Karim Hanna, Yang Gong, Zhan Zhang, Xiao Luo, and Mia Liza A Lustria. 2023. Annotation and information extraction of consumer-friendly health articles for enhancing laboratory test reporting. In *AMIA Annual Symposium Proceedings*, volume 2023, page 407. American Medical Informatics Association.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Andrea Padoan and Mario Plebani. 2022a. Artificial intelligence: is it the right time for clinical laboratories? *Clin Chem Lab Med*, 60(12):1859–1861.

Andrea Padoan and Mario Plebani. 2022b. Flowing through laboratory clinical data: the role of artificial intelligence and big data. *Clinical Chemistry and Laboratory Medicine (CCLM)*, 60(12):1875–1880.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Meru A Patil, Sandip Bhaumik, Soubhik Paul, Swarupananda Bissoyi, Raj Roy, and Seungwoo Ryu. 2013. Estimating personalized risk ranking using laboratory test and medical knowledge (umls). In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1274–1277. IEEE.

Mario Plebani. 2009. Interpretative commenting: a tool for improving the laboratory–clinical interface. *Clinica Chimica Acta*, 404(1):46–51.

Emma Stevenson, Chelsey Walsh, and Luke Hibberd. 2024. Can artificial intelligence replace biochemists? a study comparing interpretation of thyroid function test results by chatgpt and google bard to practising biochemists. *Annals of Clinical Biochemistry*, 61(2):143–149.

Tao Tu, Anil Palepu, Mike Schaekermann, Khaled Saab, Jan Freyberg, Ryutaro Tanno, Amy Wang, Brenna Li, Mohamed Amin, Nenad Tomasev, et al. 2024. Towards conversational diagnostic ai. *arXiv preprint arXiv:2401.05654*.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerová, et al. 2024. Adapted large language models can outperform medical experts in clinical text summarization. *Nature medicine*, 30(4):1134–1142.

Youfu Yan, Yu Hou, Yongkang Xiao, Rui Zhang, and Qianwen Wang. 2024. Knownet: Guided health information seeking from llms via knowledge graph integration. *IEEE Transactions on Visualization and Computer Graphics*.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Cyril Zakka, Rohan Shad, Akash Chaurasia, Alex R Dalal, Jennifer L Kim, Michael Moor, Robyn Fong, Curran Phillips, Kevin Alexander, Euan Ashley, et al. 2024. Almanac—retrieval-augmented language models for clinical medicine. *NEJM AI*, 1(2):AIoa2300068.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Appendix

A.1 The KG Schema

Start Node Type	Relationship Type	End Node Type
TestResult	CONTRIBUTES_TO	Condition
Condition	COMPOSED_OF	StandardTerm
Report	HAS_PATIENT	Patient
Patient	HAS_TEST	Test
Patient	HAS_COMMENT	Comment
Patient	HAS_CONDITION	Condition
Patient	HAS_AGE	Age
Patient	HAS_GENDER	Gender
Patient	HAS_RESULT	TestResult
Condition	IS_A	StandardTerm
Test	HAS_REF	Reference
Test	HAS_UNIT	Unit
Test	HAS_TEST_RESULT	TestResult

Table 4: The schema of the lab-kg

A.2 An Query Example

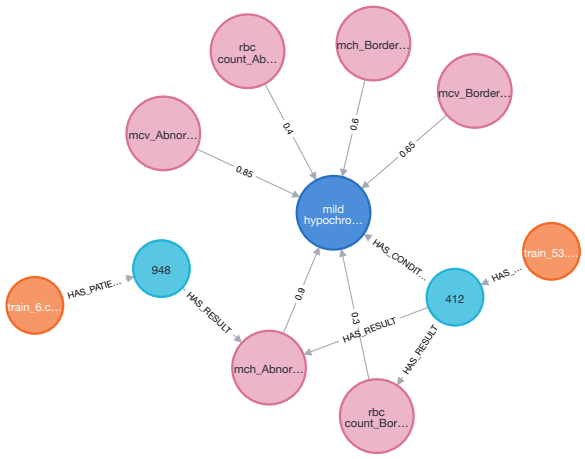

Figure 5: The "mild hypochromia" query process for a new patient. The query patient with ID 948 lacks "RBC count Borderline Abnormal (Low)" compared to the example patient with ID 412 using strict test result matching. The RBC count is not necessary for "mild hypochromia".

A.3 Confidence Score Calculation

In this example, the weights associated with the condition "mild normochromic normocytic anaemia" are shown in Table 2.

The maximum possible score is the sum of the highest weights for each test result:

$$\begin{aligned}\text{Max Score} &= 0.95 + 0.95 + 0.9 + 0.85 \\ &\quad + 0.7 + 0.5 + 0.4 + 0.3 \\ &= 5.55\end{aligned}$$

We sum the weights of the matching test results for the patient's test results, omitting borderline weights if there is a corresponding abnormal weight with a higher value. In this case, the confidence score is:

$$\begin{aligned}\text{Patient Score} &= 0.75 + 0.75 + 0.85 \\ &\quad + 0.5 + 0.4 + 0.3 \\ &= 3.55\end{aligned}$$

The normalised confidence score is:

$$\text{Confidence Score} = \frac{3.55}{5.55} = 0.64$$

A.4 KG and KG * Comparison

To compare the differences between the results from *CONTRIBUTES_TO* relationships built by the LLM with and without examples, we compared the generated results and highlighted the most significant differences below:

1. **KG with relevance built with example wins**

   KG * cannot generate "Mild absolute neutropenia" in many cases, which is in the target output. The reason is that the LLM only assigns weight to *neutrophils percentage_Abnormal (Low)*, *absolute neutrophils count_Borderline (Low)*, and *neutrophils percentage_Borderline (Low)*. However, the absolute neutrophil count comprises *absolute segmented neutrophil count* and *absolute band neutrophil count*, which are present in the examples but not recognised by the LLM's internal knowledge without examples.

2. **KG with Relevance Built without Example Wins**

   The KG built with examples cannot use strict match for the condition "mild hypochromia." Because some example patients (12.5% across all the patients with that condition) have a test result *RBC count_Abnormal (High)* associated with that condition, the LLM connects that test result to the condition with a low weight (0.4). However, in the

strict match, the new patient only has the test result *MCV_Abnormal (Low)*, which is not present in the example patient who has both *MCV_Abnormal (Low)* and *RBC count_Abnormal (High)*. The internal LLM did not connect *RBC count_Abnormal* to that condition, and it can retrieve that condition by strict match. The LLM's decision is affected by the noise of the dataset, which causes this misidentification. The graph for this query is Figure 5 in appendix.

A.5 Correlation between F1 Score and Other Metrics

metrics	medcon	bleu	bert_score	meteor	rouge1	rouge2	rougeL
corr	0.95	0.75	0.94	0.91	0.91	0.9	0.85
corr *	0.97	0.66	0.91	0.87	0.84	0.9	0.74

Table 5: The correlation between F1 score and each metric, for the result built by 1. KG (corr) and 2. KG * (corr *).