DOSSIER: Fact Checking in Electronic Health Records while Preserving Patient Privacy

Haoran Zhang[†] HAORANZ@MIT.EDU
Massachusetts Institute of Technology

Supriya Nagesh NSUPRIY@AMAZON.COM
Amazon

Milind Shyani MSHYANI@AMAZON.COM
Amazon

Nina Mishra NMISHRA@AMAZON.COM
Amazon

Abstract

Given a particular claim about a specific document, the *fact checking* problem is to determine if the claim is true and, if so, provide corroborating evidence. The problem is motivated by contexts where a document is too lengthy to quickly read and find an answer. This paper focuses on electronic health records, or a medical dossier, where a physician has a pointed claim to make about the record. Prior methods that rely on directly prompting an LLM may suffer from hallucinations and violate privacy constraints. We present a system, DOSSIER, that verifies claims related to the tabular data within a document. For a clinical record, the tables include timestamped vital signs, medications, and labs. DOSSIER weaves together methods for tagging medical entities within a claim, converting natural language to SQL, and utilizing biomedical knowledge graphs, in order to identify rows across multiple tables that prove the answer. A distinguishing and desirable characteristic of DOSSIER is that no private medical records are shared with an LLM. An extensive experimental evaluation is conducted over a large corpus of medical records demonstrating improved accuracy over five baselines. Our methods provide hope that physicians can privately, quickly, and accurately fact check a claim in an evidence-based fashion.

1. Introduction

Medical professionals benefit from Electronic Health Records (EHRs) in many ways including improved clinical care (Manca, 2015) and accurate predictive systems (Henry et al., 2022). However, for higher risk patients, these EHRs can be hundreds of pages long, making navigation time-consuming (Holmes et al., 2021; Downing et al., 2018; Overhage and McCallie Jr, 2020). One prior study found that ICU clinicians encountering new patients spend an average of 15 minutes reviewing the EHR for a typical case, and 25 minutes for a complex case. Furthermore, 49% of the clinicians reported that their chart review workflow was disorganized, with too many total data elements to review (Nolan et al., 2017). This paper's objective is to design methods to improve a medical professional's operational efficiency, specifically chart review time.

[†] Work done during an internship at Amazon.

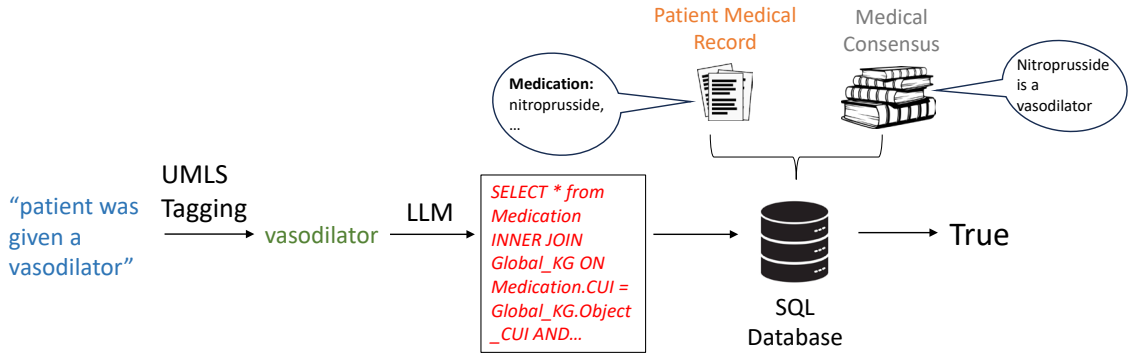


Figure 1: **Dossier**: Given a claim, we first extract clinical entities, and map them to the Unified Medical Language System (UMLS). We then use a Large Language Model (LLM) to generate a logical query, which is run against the evidence and a global knowledge base to retrieve relevant evidence and determine if the statement is correct. Crucially, the SQL query can only retrieve real patient records, so the impact of any hallucinations is reduced compared to directly passing patient records into an LLM. Note that a simple claim is shown here for demonstration purposes, and a more complex claim can be found in Table C.1.

We frame the clinical task in the language of *fact checking* (Guo et al., 2022). Given a claim such as "The patient took blood thinners within the past 12 hours", the goal is to determine if the answer is "True", "False" or there is "Not Enough Information" (NEI). Moreover, if the answer is True or False, the solution must provide corroborating evidence. For example, the answer could be "True, the patient was given 5mg of warfarin at 10am" (warfarin is a blood thinner). If no evidence of a blood thinner is found, then the answer is NEI, since there may be activity not recorded in the data.

Claims What is a realistic claim? First, note that the universe of possible claims associated with a clinical record can be quite large. Consequently, we focus on claims that can be answered with tables associated with a medical record. Examples of tables include timestamped vital signs, labs, medications, and procedures. Our medical dossier is comprised of these tables. To identify a set of realistic claims that could be made about these tables, we appeal to past work such as Lee et al. (2022); Lehman et al. (2022) which document real physician and nursing questions. These real questions motivate the templates used to generate our claims.

Considerations What is a good way to fact check a claim? Given the impressive success of LLMs in the medical domain (Thirunavukarasu et al., 2023; Singhal et al., 2023), it is natural to wonder if they can already solve the fact-checking problem. After all, one can already prompt an LLM with a question about a patient and their medical record (Agrawal et al., 2022). However, there are many challenges including: cost, hallucinations and privacy. (1) A significant challenge is cost. LLMs today charge by the token. If a long input prompt is sent to an LLM each time a claim needs to be fact checked, significant cost may be incurred to the hospital. In a hospital or ICU setting, each patient could have thousands of measurements and events, greatly increasing the LLM cost. (2) LLMs are notorious for hallucinations (Umapathi et al., 2023). For example, prior work has shown that LLMs can propagate the spread of misinformation related to the use of Ivermectin to treat COVID-19 (Vykopal et al., 2023). Thus, a fact-checking solution built exclusively on

LLMs may not be trusted. (3) Privacy is a major consideration with patient records. We cannot input a medical record into an LLM which may later use this data for training. Even publicly available datasets such as MIMIC can only be used with certain cloud providers. For a hospital, the privacy requirement is more stringent and may require an IRB approval as well as first removing personally identifiable information.

How can we prove a claim? Our objective is to design a solution where a "proof" (actual rows of the tables) is returned when the claim is True or False. Therefore, we build upon the extensive literature on text-to-SQL. Given a natural language claim and the column headings of the tables, an LLM is used to generate a SQL query. The returned rows of the SQL query constitute the desired proof of the claim. Also, restricting the use of an LLM to generate SQL limits hallucination to SQL code. Even if a physician does not understand SQL, the returned rows can be scanned for accurate interpretation of the claim.

Dossier We introduce a system, Dossier – Domain-Specific, Text-to-Sql, Semantic Fact Checker) – that addresses these limitations. One of the distinguishing aspects of Dossier is that clinical records are never sent to an LLM. Instead, a natural language claim is converted to a SQL query by first extracting medical concepts from the claim using medical taggers and then providing these concepts together with the database schema to an LLM to infer a SQL query.

Another important and novel component of Dossier is its combined local and global knowledge. The local knowledge is derived from a patient's record and represents facts specific to the patient. The global Knowledge Graph (KG) provides the source of ground truth for general medical information, and contains triples such as ("warfarin", "is a", "blood thinner"). The merged graphs constitute the database of facts. Combining these two sources of information allows our system to answer complex questions requiring multiple patient tables and general clinical knowledge.

The SQL query generated by an LLM is run against the database of facts. The evidence that the claim is True or False corresponds to the rows returned from the SQL query. Note that while LLMs are used in our solution, the hallucination is restricted to a step of the fact checking process, namely, text to SQL. For an overview of Dossier, see Figure 1.

Contributions We combine locals EHRs with a global KG to enable fact checking for highly specialized and time-sensitive claims in clinical healthcare. We build a trustworthy fact verification pipeline by synergizing the power of LLMs, structured data, and specialized knowledge graphs. Crucially, no clinical data is shared with an LLM in our method. To summarize, we make the following contributions:

1. We demonstrate that integrating local and global knowledge graphs enables us to effectively fact-check claims that require external knowledge. From the last column of Table 1, we find that incorporating the knowledge graph increases accuracy from 55.0% to 63.1% for Claude-2.
2. We find that the use of medical taggers to identify entities within a claim significantly enhances accuracy. Specifically, when combined with a global knowledge graph, we observe an improvement in accuracy from 55.0% to 75.1%.
3. One of the striking findings of this paper is that the risk of privacy, cost and hallucination associated with sharing a medical record with an LLM is unnecessary. One can more

accurately fact check using just the column headings of the tables and a text-to-SQL engine, together with medical knowledge. Concretely, this strategy far outperforms directly prompting a local LLM with retrieved rows of a patient's EHR tables (e.g. 75.1% vs. 37.3% for MedAlpaca 7B in Table 1).

4. We use a template-based approach to create a dataset containing 4,250 realistic clinical claims on individual EHRs from real-world ICU patients. Dossier outperforms prior work (Lee et al., 2022) by 24.1% on this dataset.

Generalizable Insights

1. **Text-to-SQL for fact checking is better than providing patient records directly into the LLM.** One way to use LLMs to answer questions or verify facts about tables is by providing the table and the claim within the context of the LLM, potentially with a first retrieval stage. In order to maintain patient privacy, this restricts us to use only local clinical LLMs such as ClinicalCamel (Toma et al., 2023). We find that this approach is less accurate by 45.7% on average.

   Instead, we claim that a better approach is to translate a natural language question to SQL using modern text-to-SQL approaches. This allows us to use more powerful API-based LLMs such as Claude (Anthropic, 2023b). The SQL query generated is used to verify the fact on a local machine containing the EHRs.

2. **Fear of LLM Hallucination.** There is a general fear of LLM hallucination in the community (Umapathi et al., 2023; Pal and Sankarasubbu, 2024), especially among physicians. While LLMs can hallucinate, by focusing the LLM on a narrower task of text-to-SQL (vs. open-ended question answering), we can mitigate the extent of the hallucination. Among the SQL queries generated by our method, less than 3% are not executable.

3. **Combining LLMs with domain-specific facts improves fact checking accuracy.** While LLMs may have "read everything on the internet", it may not know when or how to apply what it has read (Zhao et al., 2024). In our case, combining the LLM with databases of domain-specific knowledge simplifies the SQL query and improves the fact checking accuracy by 19.3%. In the healthcare domain, there are publicly available resources such as knowledge graphs and UMLS (Bodenreider, 2004), which contain such domain-specific facts.

2. Related work

Fact checking is a widely studied problem. Existing methods can be broadly categorized into neural and symbolic methods (Guo et al., 2022).

Neural Fact Checking. Neural methods rely on a large neural network with several connected components to detect a claim, find evidence, take a stance and justify it. For a given claim, evidence is retrieved using keyword or neural search over some ground truth corpus such as Wikipedia. The combined claim and evidence is given to a natural language inference model or a large language model (LLM) to classify as true, false or not enough evidence (Martín et al., 2022; Sathe and Park, 2021; Atanasova et al., 2020; Kotonya and Toni,

2020). Such methods have been recently re-popularized with the rise of retrieval augmented generation (RAG) and powerful in context learning abilities of LLMs (Logan et al., 2019; Lewis et al., 2020; Lazaridou et al., 2022; Wang et al., 2024). Several fact-checking datasets have been released in recent years to benchmark the performance of fact checking systems (Akhtar et al., 2022; Aly et al., 2021; Chen et al., 2019). However, most datasets are based on Wikipedia or general web corpora, i.e. data distributions that are part of the training corpora of LLMs, and do not provide a robust measurement of the fact-checking abilities of LLMs on out of distribution datasets.

But even for in-distribution datasets, the most well trained models are still prone to hallucinations or biases (Bang et al., 2023). Mitigating LLM hallucinations is an open problem, and there is a growing literature studying it, especially in the context of reasoning and fact-checking (Li et al., 2022; Dai et al., 2022). One way to mitigate hallucinations is to use knowledge bases during language model pre-training (Yu et al., 2022; He et al., 2021) or inference (Wang et al., 2021; Ke et al., 2021; Yasunaga et al., 2021; Peng et al., 2023; Pan et al., 2023a). However, the absence of reliable guarantees (theoretical or empirical) about the truthfulness of LLMs, makes it hard to deploy them off-the-shelf in critical fields such as healthcare.

Symbolic Fact Checking. Symbolic methods utilize the high quality structured information present in knowledge bases such as knowledge graphs (KG) or relational databases. A given claim is first converted into a structured query (such as SQL or SPARQL), using a neural network or a deterministic parser, which is then executed against a KG or a database (Shiralkar et al., 2017; Lin et al., 2019; Gad-Elrab et al., 2019; Wang et al., 2020a; Park et al., 2021). Symbolic methods lead to trustworthy results, but are limited by the knowledge of the database, and the power of the neural network or the deterministic parser. Recently, many methods to enhance the knowledge of KGs using language models have been proposed (Pan et al., 2023b; Baek et al., 2023; Xie et al., 2022; Zhang et al., 2020), but they are in turn limited by the truthfulness of language models.

Most relevant to our work is research on query-based EHR question answering (Wang et al., 2020a; Park et al., 2021; Lee et al., 2022). These methods often utilize an underlying text to SQL engine to convert natural language physician questions into structured SQL queries. Text to SQL conversion has been a significant challenge in natural language processing, attracting substantial interest from the research community (Yu et al., 2019; Deng et al., 2022). Recent initiatives have utilized text to SQL engines for applications such as fact-checking (Jo et al., 2019) and reducing hallucinations in LLMs (MyScale, 2023).

Our method builds upon a text to SQL framework but introduces two novel components. First, we adapt a long line of work which propose and evaluate methods for biomedical concept tagging (Kraljevic et al., 2021; Reátegui and Ratté, 2018; Savova et al., 2010) to the problem of fact-checking natural language claims on tabular EHR data. Second, We propose to fuse local and global knowledge graphs. To the best of our knowledge, no prior work has combined biomedical knowledge graphs with tabular EHRs for the purpose of fact checking or question answering. We empirically demonstrate that the addition of these two components improves performance on our clinical dataset by a significant margin. Our clinical dataset draws inspiration from the templates and real physician and nursing questions found in Lee et al. (2022). Additionally, our dataset includes new templates that necessitate external

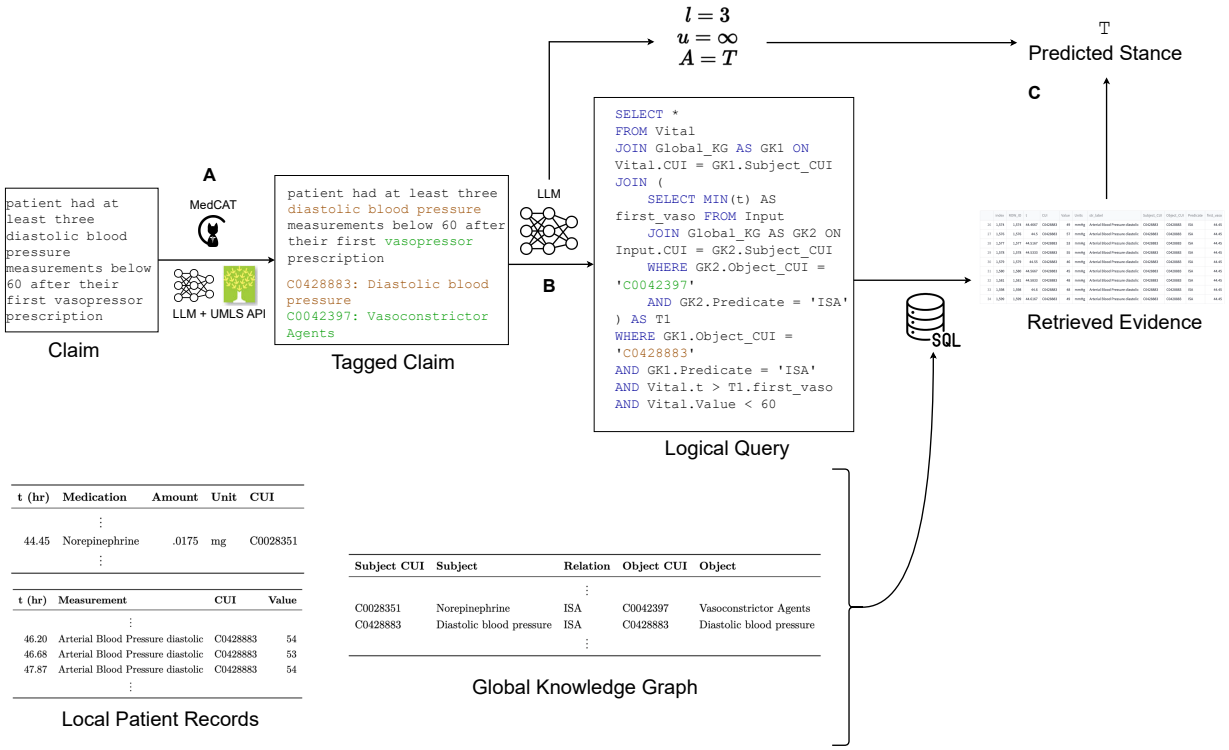


Figure 2: A complex sample claim verified by our pipeline Dossier. (A) We begin by tagging the claim with its UMLS entities. (B) The tagged claim is passed to an LLM, which generates the logical query as well as some metadata. This logical query is run on a SQL database containing the local patient records and the global knowledge graph. (C) The retrieved evidence, as well as the previously generated metadata, is used to determine the stance of the claim. For an explanation of why ablated versions of our pipeline fail on this claim, see Table C.1.

knowledge for resolution. We find that integrating knowledge graphs enables us to effectively address these complex questions.

3. Methods

In this section, we define the precise problem statement and outline the methods employed to address it. Our approach is motivated by three central observations: 1) The performance and knowledge of LLMs deteriorate in highly specialized domains. To counteract this, LLMs require assistance from specialized, curated resources, such as biomedical knowledge graphs. 2) Domain-specific and jargon-heavy statements can greatly benefit from a highly specialized entity tagging system that accurately identifies and categorizes key terms and concepts. 3) Implementing Text-to-SQL effectively constrains the hallucinations of LLMs to the query generation phase alone, while still allowing it to interact privately with the health record. Additionally, this approach helps maintain patient privacy and reduces costs.

3.1. Problem Setup

In the fact checking problem, we are given a natural language claim C, made at time $t_C > 0$, and a set of documents $E = \{d_1, ..., d_k\}$, each with timestamp t_{d_i}, and where $t_{d_i} < t_C \ \forall i$. We are also given a stance label $Y \in \{\textbf{T}\text{rue}, \textbf{F}\text{alse}, \textbf{N}\text{ot Enough Information}\}$. Here, d_i is a lab, vital measurement or drug prescription. We are also given a static global knowledge graph G of biomedical information. The goal is to build a system Ω such that $\Omega(C, E, G) = (\hat{Y}, \hat{E})$, where $\hat{Y} \in \{\texttt{T}, \texttt{F}, \texttt{N}\}$, and $\hat{E} \subset E$. When $\hat{Y} \in \{\texttt{T}, \texttt{F}\}$, we would like $\hat{E}$ to be the evidence that "proves" $\hat{Y}$. If no such $\hat{E}$ exists, then we should return $\hat{Y} = \texttt{N}$.

3.2. Our Approach

We determine the factual correctness of a given claim by turning it into a logical query that can be evaluated against the evidence and global knowledge. Our method assumes that the evidence and global knowledge are represented in a structured format. In our work, the evidence are the patient EHRs, which are in the form of tables. In a hospital setting, this could easily correspond to thousands of vital measurements, hundreds of lab measurements, and tens of input medications per patient. Our method (Figure 1) involves the following steps. See Figure 2 for a full example.

Global KG: For a query such as "patient was given a blood thinner", the text to SQL engine will not be able to write a query that includes all blood thinning drugs by itself. The presence of the global KG circumvents this problem by providing a structured repository of biomedical information that is utilized by the engine during query time.

For the global knowledge graph G, we utilize SemMedDB (Kilicoglu et al., 2012), a repository of biomedical knowledge derived from the abstracts of PubMed articles. The knowledge graph G comprises a collection of triples $\{(e_i, r_{i,j}, e_j)\}$, where e_i and e_j are entities such as *warfarin* and *blood thinner*, respectively, and r_{ij} denotes a relation, such as *is a*. We filter SemMedDB to select 7.3 million edges, each of which have at least 15 references.

Domain specific entity tagging: A clinical concept could have many unique names. For example, "blood thinners" and "anticoagulants" refer to the same clinical concept, and either one could be used in a clinical claim. The first step in our pipeline is to identify the entities referenced in the claim, and map them to a common vocabulary. The common vocabulary we use is the Unified Medical Language System (UMLS) (Bodenreider, 2004), which contains a Concept Unique Identifier (CUI) for each clinical concept. We use two methods to tag UMLS entities from a claim: (1) MedCAT (Kraljevic et al., 2021), and (2) an LLM for clinical entity extraction, followed by the UMLS API to convert it to a CUI. We use the union of the entities tagged by each of these methods.

Logical query generation: Once we have the entities extracted, we translate the claim into a logical query. Given only the tagged biomedical entities, and the schema of the EHR and the global KG, we use an LLM to convert the claim into a SQL query. Crucially, the LLM does not need access to the patient records, only the table schemas.

Stance determination: The query that is generated by the LLM is run on a SQL database containing the evidence and global knowledge graph, to retrieve $\hat{E}$. When $\hat{E}$ is empty, we return $\texttt{N}$. If $\hat{E}$ is non-empty, then this becomes the supporting evidence. For example, if the

claim is: "patient was given a vasodilator at least twice", the generated query extracts the rows from the patient's medications table where a vasodilator was administered. To decide whether we should return `T` or `F`, we compare $|\hat{E}|$ with a numeric interval output by the LLM. In this example, the interval is defined by the lower bound $l = 2$ and upper bound $u = \infty$ (more details in Appendix B). When the query fails to run, or when the LLM does not return a SQL query, we also return `N`.

4. Experiments

4.1. Experimental setup

To validate our proposed pipeline, we devise 20 expressive templates (see Appendix D), which are used to generate 4,250 claims for 100 randomly selected admissions from MIMIC-III (Johnson et al., 2016) over four patient tables: ADMISSIONS, LABEVENTS, CHARTEVENTS and INPUTEVENTS. Details on data processing can be found in Appendix A.

While our templates are inspired by the real physician and nursing queries released in Lee et al. (2022), the dataset differs in that it focuses on individual patient records and includes queries that require global knowledge for verification. Among the queries that necessitate a Knowledge Graph (KG), "patient was given a vasodilator" is a representative example. The patient's EHR in INPUTEVENTS might list "Lisinopril/Nitroprusside" but not "vasodilator," thus necessitating a global KG connecting the two entities for verifying such claims.

We also organize our templates in increasing order of difficulty during evaluation. The difficulty of a given template depends on the number of tables required to answer it and whether it requires external medical knowledge (i.e. via the global KG). Reasoning over multiple tables while using the global KG presents the highest difficulty. We believe that this organization provides a natural hierarchy of difficulty that can be used to measure the performance of fact-checking systems with increasingly complex claims and datasets. Our proposed method achieves state-of-the-art performance on almost all difficulty levels.

For each template, we manually construct a gold-standard SQL query which is run on each patient's EHR to determine the true label. We evaluate our pipeline on each claim, generating SQL queries (with temperature = 0) using three LLMs: Claude-2 (Anthropic, 2023b), Claude-instant-1 (Anthropic, 2023a), and CodeLlama-13B (Rozière et al., 2023). The choice of these models is based on recent benchmarks which find that API-based LLMs such as GPT-4 and Claude-2 are currently state-of-the-art in the text-to-SQL task (Li et al., 2023). In addition, we include CodeLlama as an example of a publicly available open-source model. We provide four in-context examples, and the prompts used can be found in Appendix E.

4.2. Baseline methods

As a baseline approach, we pass patient records directly into open-source LLMs along with the claim to be fact checked. Open-source LLMs, which are freely and publicly available, can be downloaded and used locally, thus circumventing the issue of patient privacy. We use powerful open-source LLMs that have been trained on medical or clinical data: ClinicalCamel 13B (Toma et al., 2023), MedAlpaca 7B (Han et al., 2023), and Asclepius 13B (Kweon et al., 2023). As the string representations of these records almost always exceed the LLM's context length, we use BM25 (Robertson et al., 1994) or semantic similarity (Wang et al., 2020b)

retrieval, matching on the measurement name, to retrieve the most relevant records. We also incorporate a Llama-2 7B model (Touvron et al., 2023) as a baseline, featuring a large context length (32k), where we pass the entire record directly. Since the above models have access to patient health records, they do not need to execute an SQL query and instead infer the answer by reasoning in-context. Finally, we also include as baseline a T5-Base (Raffel et al., 2020) model trained on the EHR-SQL dataset (Lee et al., 2022).

4.3. Ablations

We consider the following ablations of the DOSSIER pipeline (See Table C.1 for an example):

- **Only UMLS**: We first ablate the global KG. In this setup, the SQL query no longer has access to a global knowledge source.
- **Only Global KG**: By ablating the UMLS tagging, the SQL queries have to search over the patient records (or the global KG) by string matching.
- **Neither**: By ablating both components, the SQL query can only use string matching on local patient tables.

5. Results

We start with an overall accuracy comparison, adding and removing components of DOSSIER in order to quantify the impact. We evaluate performance on specific subsets of the dataset based on difficulty as mentioned earlier. Then, to dig into DOSSIER's mistakes, we categorize the model errors with manual annotation. Following that, the impact of claim rephrasing is explored. Finally, we evaluate performance on templates based on real physician and nursing queries in Lee et al. (2022). For baselines with direct EHR prompting, we show results using BM25 retrieval in the main text, and results for retrieval with all-MiniLM-L6-v2 (Wang et al., 2020b) can be found in Table C.6.

5.1. Overall Performance

In Table 1, we compare the performance of all methods on the generated claims. We find that our pipeline with Claude-2 far outperforms the baselines. We find that ablating the components of our pipeline hurts the model for all difficulty levels except one (no global KG with 2 tables). In some cases, adding only UMLS tagging and not the global KG can actually hurt performance compared to having neither component. We explore these results further by showing the percentage of times that the model explicitly says that it does not know the answer in Table C.3, and the percentage of times that the resulting query gives an error in Table C.4.

To examine the trend of ablating each component, we plot the accuracy for the most difficult claims – those which require a global KG and two local tables – in Figure 3. T5-EHRSQL achieves the best performance of all baselines at 52.0%. The accuracy of the remaining baseline approaches never exceeds 40%. On the other hand, DOSSIER with Claude-2 achieves an accuracy of 75.1%. We also evaluate the quality of retrieved evidence for each claim. We focus on the Claude-2 full pipeline, and manually examine 100 claims which have $Y \in \{\texttt{T}, \texttt{F}\}$, and for which the pipeline predicts the correct stance. We find that in 100% of

Table 1: Accuracy (%) of clinical fact-checking methods evaluated on 4,250 claims generated from MIMIC-III, stratified by whether the claim requires the Global KG, and the number of local patient tables required. "Full" represents our full pipeline, and we also provide ablations for two components of our pipeline. Best performances for each LLM are **bolded**, and the best overall performances are shaded in gray.

	Requires Global KG	**No**		**Yes**	
	# Local Tables	**1**	**2**	**1**	**2**
Dossier	**Claude-2 Full**	**84.2**	74.7	**68.9**	**75.1**
	Only UMLS	82.5	73.1	52.0	53.4
	Only Global KG	66.6	48.3	61.8	63.1
	Neither	79.3	**82.6**	50.2	55.0
	Claude-1 Full	**79.2**	57.9	**64.8**	**63.2**
	Only UMLS	77.4	**64.8**	51.0	53.6
	Only Global KG	65.0	48.1	58.6	59.2
	Neither	74.1	63.3	50.4	54.2
	CodeLlama 13B Full	**75.3**	51.0	**66.4**	**53.6**
	Only UMLS	72.9	**53.6**	49.1	52.1
	Only Global KG	58.7	47.1	48.5	52.3
	Neither	64.5	47.7	49.3	51.7
Baselines	**MedAlpaca 7B**	52.8	44.6	41.5	37.3
	ClinicalCamel 13B	37.5	27.3	27.3	28.2
	Asclepius 13B	47.2	26.6	34.1	33.5
	Llama2 7B 32k	29.7	31.5	46.6	31.3
	T5-EHRSQL	59.7	48.0	47.9	52.0

cases, the evidence retrieved properly supports the pipeline's prediction. Additional results can be found in Appendix C.

5.2. NEI Accuracy

In addition to overall accuracy, we also want to ensure that the method is accurate for each class "True", "False" and "NEI". We start by exploring NEI accuracy in Figure 4. Interestingly, Llama-2 shows near 0% accuracy at predicting NEI. Exploring further in Figure C.1(b), it is due to the fact that Llama-2 almost always predicts T or F. Another interesting observation is T5-EHRSQL exhibits the opposite behavior in that it almost always predicts NEI, as seen in Figure C.1(a). This is likely due to the fact that the pretrained model has not seen questions similar to the ones in our dataset, and so the resulting SQL query is much more likely to give an error (Table C.4) or retrieve no rows. On the other hand, Claude-2 appears to strike a better balance achieving high NEI accuracy, while also predicting True/False accurately as we describe next.

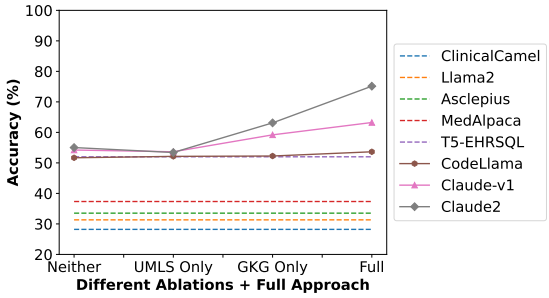


Figure 3: Accuracy of ablations and the full method, for claims which require a global KG and two local tables. Several ablations are considered depending on whether a Global Knowledge Graph (GKG) is used and whether UMLS medical tagging is used. The baseline approaches are shown as flat lines as they do not benefit from the DOSSIER pipeline. Among the methods that benefit from the DOSSIER pipeline, Claude-2 performs the best across all ablations and in full.

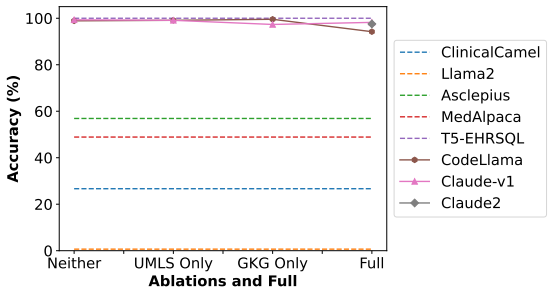
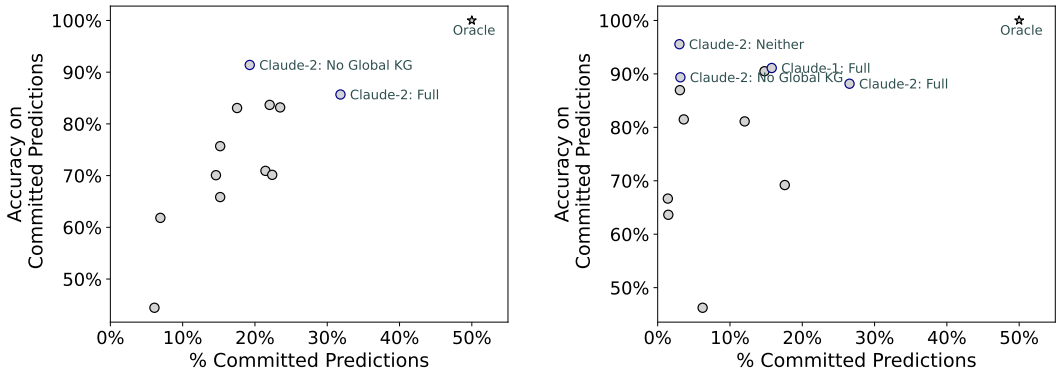


Figure 4: Accuracy on samples with label NEI. One baseline approach, Llama-2, never outputs NEI. In other words, Llama-2 outputs True or False, even when there is no evidence to support the claims. Another baseline approach, T5-EHRSQL, always outputs NEI – even where there exists evidence to prove or disprove the claim. DOSSIER labels NEI claims more accurately. Neither the global knowledge graph nor UMLS help with NEI.



(*a*) All claims (*b*) Claims that require the global KG

Figure 5: We define a "committed" prediction as one where $\hat{Y} \in \{\texttt{T}, \texttt{F}\}$, and view predicting $\hat{Y} = \texttt{N}$ as a form of deferral to the clinician for manual verification. We plot the accuracy of DOSSIER when the model commits, versus the percentage of the time it does, for (a) all claims in our dataset, and (b) the subset of claims which require the global KG. We highlight models on the pareto front, and other select models for clarity. We present the full plot with all models labelled in Figure C.2. Note that the dataset of claims consists of 50% N, 40% T, 10% F.

5.3. Accuracy on Committed Predictions

Here, we consider the fact that all errors may not have equal cost. When $\hat{Y} = \texttt{N}$, the lack of evidence should prompt the physician to manually check the claim. However, when $\hat{Y} \in \{\texttt{T}, \texttt{F}\}$, any errors would indicate the retrieval of bogus evidence, which may mislead the physician into taking an improper action. Hence, we care most about being correct when $\hat{Y} \in \{\texttt{T}, \texttt{F}\}$, and we refer to this as the model "committing" to a prediction. Thus, predicting $\hat{Y} = \texttt{N}$ can be seen as a type of deferral (Madras et al., 2018; Geifman and El-Yaniv, 2017) for manual examination by the clinician. In Figure 5, we plot the accuracy on committed samples for Dossier models and its ablations, versus the percentage of time the model commits. When examining the set of all claims (Figure 5a), we find that the full pipeline with Claude-2 commits about 10% more than when the global KG is ablated, with a small loss on the accuracy. When further subsetting to the set of claims (Figure 5b) that require a global KG, we find that the pipeline with ablated KG almost never commits, and that the full pipeline still commits the most frequently, with a minimal loss on accuracy.

5.4. Error Analysis

Here, we seek to examine incorrect predictions made by our pipeline, in order to derive insight into which component is responsible. We randomly select 50 examples for which Dossier with full Claude-2 makes an error, and manually annotate these examples for the cause of error. We present our analysis in Table 2. We find that out of the 50 errors, 40% are caused by inadequacies in the LLM for SQL generation, e.g. the generated SQL contains logical or syntactical errors, or the LLM selects an inappropriate tagged entity in its query. In addition, we find that issues with the UMLS entity taggers (i.e. MedCAT and the UMLS API) are responsible for 34% of model errors. Thus, improvements in these two components (text-to-SQL and UMLS entity tagging) are two promising primary directions to improve the performance of our pipeline.

5.5. Robustness to Paraphrasing

One common critique of template-generated claims is that they may be unrealistic, and that their syntax may be inconsistent with the claims that a physician would make (Lehman et al., 2022). Here, we evaluate the robustness of Dossier to paraphrasing of claims. We randomly select 1,000 samples from our dataset, and use Claude-2 to provide a paraphrase of the claim, using the prompt in Appendix E, emphasizing that claim semantics should not be altered. We run the 500 most realistic paraphrased claims through Dossier with Claude-2. We present our results in Table 3. We find that all models experience a decrease in accuracy on the paraphrased dataset, with the full pipeline showing a drop of 5.6%. However, the full pipeline still outperforms its ablations.

5.6. Performance on templates based on real physician & nursing queries in Lee et al. (2022)

The templates utilized in our study draw inspiration from queries documented in EHR-SQL (Lee et al., 2022). It is important to note that EHR-SQL excludes queries that are ambiguous or necessitate external knowledge. Consequently, our templates, as detailed in Appendix D,

Table 2: We randomly select 50 examples for which Dossier with Claude-2 Full makes an incorrect prediction. We manually annotate these examples to determine the source of error, and group them into the following categories, which cover 92% of observed errors. We provide an illustrative example for each source of error. The remaining 8% encompass an array of other errors, including syntax errors and issues with the output numeric interval or the attitude.

		Example	
Cause of Error	**Percentage**	**Claim**	**Description**
Entity taggers missed an entity	34%	pt was administered a Penicillin-containing product since t = 57.0	Entity taggers output the following relevant entity: `C0030824: Allergy to penicillin` Which misses C5437787: Penicillin-containing product.
Logical error in SQL generation	22%	patient had Heart Rate measurements less than 142.0 since last being administered Propofol	SQL contains incorrect operator for vital value comparison: `SELECT * FROM Vital WHERE Vital.CUI IN ( ... ) AND Vital.t > ( ... ) AND Vital.Value >= 142.0 ...`
Wrong CUI chosen from tagged list	14%	patient had Alkaline phosphatase values less than 208.0 since their last Amylase measurement greater than 117.25	From the following two tagged entities: `C0002712: Amylase` `C0201883: Amylase measurement` The LLM chose to use the former (a pharmacologic substance) instead of the latter (a laboratory procedure).
UMLS contains several CUIs for nearly identical concepts	10%	patient had Sodium values greater than 140.0 since last being prescribed .9% Normal Saline	The entity taggers return the following entities: `C0445115: Normal saline` `C0036082: Sodium chloride solution` The LLM chose to use the latter, whereas the local tables are coded with the former, and there is no edge between the two in the Global KG with an ISA predicate.
Vagueness in claim	8%	pt was given a Polysaccharides since they were last prescribed any Glycosaminoglycans	The patient was prescribed Heparin, a drug which fits into both categories. The gold SQL query used a $\geq$ comparator for time to get a label of `T`, whereas the LLM-generated query used a $>$ to predict `N`.
LLM tries to output multiple SQL queries	4%	pt did not have Glucose measurements less than 152.0 since their last PTT measurement greater than 25.2	The LLM outputs two separate SQL queries, one to select the glucose measurements, and one to select PTT.

Table 3: Accuracy (%) of DOSSIER with Claude-2, evaluated on 500 natural language claims generated by paraphrasing our templates using Claude-2.

	Original	Paraphrased
Claude-2 Full	**79.4**	**73.8**
No Global KG	70.2	68.4
No UMLS	61.6	57.8
Neither	72.6	65.8

Table 4: Accuracy (%) of clinical fact-checking methods evaluated on the claims generated using **only the templates that intersect with the templates in Lee et al. (2022)** which are derived from real physician and nursing questions.

		Accuracy (%)
DOSSIER	**Claude-2 + UMLS**	**95.69**
	Claude-1 + UMLS	86.93
	CodeLlama 13B + UMLS	77.10
Baselines	MedAlpaca 7B	52.10
	ClinicalCamel 13B	38.65
	Asclepius 13B	43.90
	Llama2 7B 32k	34.78
	T5-EHRSQL	55.25

incorporate new templates of increased complexity—requiring the use of multiple tables and a global Knowledge Graph (KG). This approach not only represents a broader range of general queries but also facilitates a systematic evaluation of future fact-checking systems based on the level of difficulty.

We benchmarked the performance of DOSSIER on a subset of our templates that are identical to those used in EHR-SQL. This subset, by design, only includes statements that do not require external knowledge, i.e., no global KG, and rely solely on one table for verification. We found that on this subset, Claude-2 with UMLS tagging significantly outperformed several baselines. This outcome underscores the significance of UMLS tagging within our pipeline and demonstrates the capabilities of modern LLMs, which were employed due to our privacy-preserving method. These results are detailed in Table 4.

6. Discussion

Performance of DOSSIER We see that the DOSSIER pipeline using Claude-2 performs the best in terms of the overall accuracy. In addition, in the setting described in this work, scenarios where a fact checking system returns NEI would be followed by manual analysis by the physician. Hence, it is crucial for the system to be accurate when it returns either `T` or `F`. We find that the DOSSIER pipeline with Claude-2 strikes a balance between committing to an answer and being accurate.

Cost We also note that DOSSIER text-to-SQL approach is generally more cost-effective. For a given query, we use approximately 2,000 tokens for inference —- 1,500 for the input prompt and fewer than 500 for the few-shot examples. In contrast, as a rough estimate, directly feeding the EHR to an LLM compromises privacy and significantly increases the number of tokens to potentially more than 50,000 tokens, assuming an EHR with 10,000 events (rows) per patient and 5 tokens per row. Given that LLM inference costs increase linearly with the number of tokens, our method can be approximately 25 times more cost-effective.

Fact checking claims in the context of EHRs In this paper, we frame the problem of fact checking claims made on a patient's EHR. The motivation being reducing the burden on medical staff manually going through lengthy EHRs to find answers to pointed questions such as "The patient was prescribed medication X". Through our experiments on the MIMIC

III dataset, we find that the Dossier system can be used to determine the factuality along with evidence from the EHR with high accuracy.

While this paper is based on fact checking claims made by physicians, LLMs are increasingly pervasive and are expected to soon play a significant role in health applications (such as question answering tasks on EHRs) (Thirunavukarasu et al., 2023; Castonguay and Lovis, 2023). As such, verifying and fact checking claims made *by LLMs* would be critical. For example, a physician may pose the question "Was the patient given blood thinners in the past 12 hours" to an LLM. In order to mitigate hallucinations, it is crucial to verify the response or claim made by the LLM. We believe that the Dossier system would be valuable in fact checking claims made by LLMs, in cases where such claims are verifiable from data within tabular EHRs. See Appendix F for further discussion.

Limitations Our method has several limitations, related to the nature of the local and global knowledge graphs, and to the use of an LLM. First, our method assumes that all knowledge is in a structured format such as a table or graph. In particular, we have not studied claims that are related to free-form patient notes. One promising approach would be to transform the free-form text into structured knowledge (e.g. with relational entity extraction) (Jain et al., 2021; Altuncu et al., 2019; Fatima et al., 2017). Next, our claims are generated by random slot filling of the templates shown in Table D.7. It is possible that we create some ambiguous or implausible claims. For example, "Patient was given vasodilator after their blood pressure measurement was below 80/50 mmHg". This is an implausible statement since vasodilators are not given to someone with a low blood pressure. While verifying our pipeline with claims made by real physicians would be the gold standard, we chose to generate claims through the slot filling process due to limitations on physician time.

Finally, our fact-checking system is only as good as the *source* of the facts. However, to the best of our knowledge, this is true of all fact-checking systems, both neural and human (Uscinski and Butler, 2013). Regardless, we emphasize that our pipeline is compatible with any biomedical knowledge graph, not just SemMedDB (Kilicoglu et al., 2012). As the quality of such knowledge graphs continue to improve (Chandak et al., 2023), so would the performance of our pipeline.

Potential Risks We do not advocate for blind deployment of these models in real-world clinical settings. Practitioners should always test such models on their data and take a myriad of other considerations into account (e.g. privacy, fairness, regulation, interpretability) before deployment (Zhang et al., 2022; Wiens et al., 2019). Misuse of such models could lead to real patient harm. In addition, we acknowledge that LLMs have likely been trained on copyrighted material, and so permission should be obtained for their use whenever necessary.

7. Conclusion

Our work shows that it is possible to verify natural language claims over patient EHRs in healthcare by effectively combining modern LLMs with specialized knowledge graphs. Answering claims with ambiguously defined ground truth, such as the free text found in clinical notes, would be an interesting direction of future work. Additionally, quantifying the improvement in operational efficiency of medical professionals using Dossier is an important next step.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. Large language models are zero-shot clinical information extractors. *arXiv preprint arXiv:2205.12689*, 2022.

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. PubHealthTab: A public health table-based dataset for evidence-based fact checking. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1–16, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.1. URL https://aclanthology.org/2022.findings-naacl.1.

M Tarik Altuncu, Erik Mayer, Sophia N Yaliraki, and Mauricio Barahona. From free text to clusters of content in health records: an unsupervised graph partitioning approach. *Applied network science*, 4:1–23, 2019.

Rami Aly, Zhijiang Guo, M. Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. Feverous: Fact extraction and verification over unstructured and structured information. *ArXiv*, abs/2106.05707, 2021. URL https://api.semanticscholar.org/CorpusID:235391052.

Anthropic. Releasing claude instant 1.2. https://www.anthropic.com/index/releasing-claude-instant-1-2, 2023a.

Anthropic. Claude 2. https://www.anthropic.com/index/claude-2, 2023b.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. Generating fact checking explanations. *arXiv preprint arXiv:2004.05773*, 2020.

Jinheon Baek, Alham Fikri Aji, Jens Lehmann, and Sung Ju Hwang. Direct fact retrieval from knowledge graphs without entity linking. *arXiv preprint arXiv:2305.12416*, 2023.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity, 2023.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

Alexandre Castonguay and Christian Lovis. Introducing the "ai language models in health care" section: Actionable strategies for targeted and wide-scale deployment, 2023.

Payal Chandak, Kexin Huang, and Marinka Zitnik. Building a knowledge graph to enable precision medicine. *Scientific Data*, 10(1):67, 2023.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, SHIYANG LI, Xiyou Zhou, and William Yang Wang. Tabfact: A large-scale dataset for table-based fact verification. *ArXiv*, abs/1909.02164, 2019. URL https://api.semanticscholar.org/CorpusID:198917339.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022. doi: 10.18653/v1/2022.acl-long.581. URL http://dx.doi.org/10.18653/v1/2022.acl-long.581.

Naihao Deng, Yulong Chen, and Yue Zhang. Recent advances in text-to-SQL: A survey of what we have and what we expect. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na, editors, *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2166–2187, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.190.

Kevin Donnelly et al. Snomed-ct: The advanced terminology and coding system for ehealth. *Studies in health technology and informatics*, 121:279, 2006.

N Lance Downing, David W Bates, and Christopher A Longhurst. Physician burnout in the electronic health record era: are we ignoring the real cause?, 2018.

Jean-Baptiste Excoffier, Tom Roehr, Alexei Figueroa, Michalis Papaaioannou, Keno Bressem, and Matthieu Ortala. Generalist embedding models are better at short-context clinical semantic search than specialized embedding models. *arXiv preprint arXiv:2401.01943*, 2024.

Arooj Fatima, Arsalan Ghazi, and Cristina Luca. Semantic graph from free-text. In *2017 International Conference on Optimization of Electrical and Electronic Equipment (OPTIM) & 2017 Intl Aegean Conference on Electrical Machines and Power Electronics (ACEMP)*, pages 1132–1137. IEEE, 2017.

Mohamed H. Gad-Elrab, Daria Stepanova, Jacopo Urbani, and Gerhard Weikum. Exfakt: A framework for explaining facts over knowledge graphs and text. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, page 87–95, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359405. doi: 10.1145/3289600.3290996. URL https://doi.org/10.1145/3289600.3290996.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30, 2017.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206, 2022. doi: 10.1162/tacl_a_00454. URL https://aclanthology.org/2022.tacl-1.11.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*, 2023.

Lei He, Suncong Zheng, Tao Yang, and Feng Zhang. KLMo: Knowledge graph enhanced pretrained language model with fine-grained relationships. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4536–4542, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.384. URL https://aclanthology.org/2021.findings-emnlp.384.

Katharine E Henry, Roy Adams, Cassandra Parent, Hossein Soleimani, Anirudh Sridharan, Lauren Johnson, David N Hager, Sara E Cosgrove, Andrew Markowski, Eili Y Klein, et al. Factors driving provider adoption of the trews machine learning-based early warning system and its effects on sepsis treatment timing. *Nature medicine*, 28(7):1447–1454, 2022.

John H Holmes, James Beinlich, Mary R Boland, Kathryn H Bowles, Yong Chen, Tessa S Cook, George Demiris, Michael Draugelis, Laura Fluharty, Peter E Gabriel, et al. Why is the electronic health record so challenging for research and clinical care? *Methods of information in medicine*, 60(01/02):032–048, 2021.

Saahil Jain, Ashwin Agrawal, Adriel Saporta, Steven QH Truong, Du Nguyen Duong, Tan Bui, Pierre Chambon, Yuhao Zhang, Matthew P Lungren, Andrew Y Ng, et al. Radgraph: Extracting clinical entities and relations from radiology reports. *arXiv preprint arXiv:2106.14463*, 2021.

Saehan Jo, Immanuel Trummer, Weicheng Yu, Xuezhi Wang, Cong Yu, Daniel Liu, and Niyati Mehta. Aggchecker: a fact-checking system for text summaries of relational data sets. *Proc. VLDB Endow.*, 12(12):1938–1941, aug 2019. ISSN 2150-8097. doi: 10.14778/3352063.3352104. URL https://doi.org/10.14778/3352063.3352104.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

Pei Ke, Haozhe Ji, Yu Ran, Xin Cui, Liwei Wang, Linfeng Song, Xiaoyan Zhu, and Minlie Huang. Jointgt: Graph-text joint representation learning for text generation from knowledge graphs. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021. doi: 10.18653/v1/2021.findings-acl.223. URL http://dx.doi.org/10.18653/v1/2021.findings-acl.223.

Halil Kilicoglu, Dongwook Shin, Marcelo Fiszman, Graciela Rosemblat, and Thomas C Rindflesch. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160, 2012.

Neema Kotonya and Francesca Toni. Explainable automated fact-checking for public health claims. *arXiv preprint arXiv:2010.09926*, 2020.

Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A Folarin, Angus Roberts, et al. Multi-domain clinical natural language processing with medcat: the medical concept annotation toolkit. *Artificial intelligence in medicine*, 117:102083, 2021.

Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, et al. Publicly shareable clinical large language model built on synthetic clinical notes. *arXiv preprint arXiv:2309.00237*, 2023.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. Internet-augmented language models through few-shot prompting for open-domain question answering, 2022.

Gyubok Lee, Hyeonji Hwang, Seongsu Bae, Yeonsu Kwon, Woncheol Shin, Seongjun Yang, Minjoon Seo, Jong-Yeup Kim, and Edward Choi. Ehrsql: A practical text-to-sql benchmark for electronic health records. *Advances in Neural Information Processing Systems*, 35: 15589–15601, 2022.

Eric Lehman, Vladislav Lialin, Katelyn Y Legaspi, Anne Janelle R Sy, Patricia Therese S Pile, Nicole Rose I Alberto, Richard Raymund R Ragasa, Corinna Victoria M Puyat, Isabelle Rose I Alberto, Pia Gabrielle I Alfonso, et al. Learning to ask like a physician. *arXiv preprint arXiv:2206.02696*, 2022.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2020.

Jinyang Li, Binyuan Hui, Ge Qu, Binhua Li, Jiaxi Yang, Bowen Li, Bailin Wang, Bowen Qin, Rongyu Cao, Ruiying Geng, et al. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *arXiv preprint arXiv:2305.03111*, 2023.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Zhenhua Dong, Chengjie Sun, Bingquan Liu, Zhenzhou Ji, Xin Jiang, and Qun Liu. How pre-trained language models capture factual knowledge? a causal-inspired analysis, 2022.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *arXiv preprint arXiv:1909.02151*, 2019.

Robert Logan, Nelson F. Liu, Matthew E. Peters, Matt Gardner, and Sameer Singh. Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1598. URL https://aclanthology.org/P19-1598.

David Madras, Toni Pitassi, and Richard Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in Neural Information Processing Systems*, 31, 2018.

Donna P Manca. Do electronic medical records improve quality of care?: Yes. *Canadian Family Physician*, 61(10):846–847, 2015.

Alejandro Martín, Javier Huertas-Tato, Álvaro Huertas-García, Guillermo Villar-Rodríguez, and David Camacho. Facter-check: Semi-automated fact-checking through semantic

similarity and natural language inference. *Knowledge-Based Systems*, 251:109265, 2022. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2022.109265. URL https://www.sciencedirect.com/science/article/pii/S0950705122006323.

Michael Mayers, Tong Shu Li, Núria Queralt-Rosinach, and Andrew I Su. Time-resolved evaluation of compound repositioning predictions on a text-mined knowledge network. *BMC bioinformatics*, 20:1–12, 2019.

MyScale. Teach your llm vector-sql. 2023. URL https://myscale.com/blog/teach-your-llm-vector-sql/.

Matthew E Nolan, Rodrigo Cartin-Ceba, Pablo Moreno-Franco, Brian Pickering, and Vitaly Herasevich. A multisite survey study of emr review habits, information needs, and display preferences among medical icu clinicians evaluating new patients. *Applied clinical informatics*, 8(04):1197–1207, 2017.

J Marc Overhage and David McCallie Jr. Physician time spent using the electronic health record during outpatient encounters: a descriptive study. *Annals of internal medicine*, 172 (3):169–174, 2020.

Ankit Pal and Malaikannan Sankarasubbu. Gemini goes to med school: Exploring the capabilities of multimodal large language models on medical challenge problems & hallucinations. *arXiv preprint arXiv:2402.07023*, 2024.

Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. Fact-checking complex claims with program-guided reasoning. *arXiv preprint arXiv:2305.12744*, 2023a.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying large language models and knowledge graphs: A roadmap, 2023b.

Junwoo Park, Youngwoo Cho, Haneol Lee, Jaegul Choo, and Edward Choi. Knowledge graph-based question answering with electronic health records. In *Machine Learning for Healthcare Conference*, pages 36–53. PMLR, 2021.

Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, et al. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Ruth Reátegui and Sylvie Ratté. Comparison of metamap and ctakes for entity extraction in clinical notes. *BMC medical informatics and decision making*, 18:13–19, 2018.

SE Robertson, S Walker, and S Jones. M. hancock-beaulieu, m., and gatford, m.(1995). okapi at trec-3. In *The Third Text REtrieval Conference (TREC-3)*, pages 109–126, 1994.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Aalok Sathe and Joonsuk Park. Automatic fact-checking with document-level annotations using BERT and multiple instance learning. In *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 101–107, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.fever-1.11. URL https://aclanthology.org/2021.fever-1.11.

Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513, 2010.

Prashant Shiralkar, Alessandro Flammini, Filippo Menczer, and Giovanni Luca Ciampaglia. Finding streams in knowledge graphs to support fact checking. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 859–864. IEEE, 2017.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. Large language models in medicine. *Nature medicine*, 29(8):1930–1940, 2023.

Augustin Toma, Patrick R Lawler, Jimmy Ba, Rahul G Krishnan, Barry B Rubin, and Bo Wang. Clinical camel: An open-source expert-level medical language model with dialogue-based knowledge encoding. *arXiv preprint arXiv:2305.12031*, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Logesh Kumar Umapathi, Ankit Pal, and Malaikannan Sankarasubbu. Med-halt: Medical domain hallucination test for large language models. *arXiv preprint arXiv:2307.15343*, 2023.

Joseph E Uscinski and Ryden W Butler. The epistemology of fact checking. *Critical Review*, 25(2):162–180, 2013.

Ivan Vykopal, Matúš Pikuliak, Ivan Srba, Robert Moro, Dominik Macko, and Maria Bielikova. Disinformation capabilities of large language models. *arXiv preprint arXiv:2311.08838*, 2023.

Ping Wang, Tian Shi, and Chandan K Reddy. Text-to-sql generation for question answering on electronic medical records. In *Proceedings of The Web Conference 2020*, pages 350–361, 2020a.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788, 2020b.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194, Mar 2021. ISSN 2307-387X. doi: 10.1162/tacl_a_00360. URL http://dx.doi.org/10.1162/tacl_a_00360.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *ArXiv*, abs/2401.04398, 2024. URL https://api.semanticscholar.org/CorpusID:266899992.

Jenna Wiens, Suchi Saria, Mark Sendak, Marzyeh Ghassemi, Vincent X Liu, Finale Doshi-Velez, Kenneth Jung, Katherine Heller, David Kale, Mohammed Saeed, et al. Do no harm: a roadmap for responsible machine learning for health care. *Nature medicine*, 25(9): 1337–1340, 2019.

Xin Xie, Ningyu Zhang, Zhoubo Li, Shumin Deng, Hui Chen, Feiyu Xiong, Mosha Chen, and Huajun Chen. From discrimination to generation: Knowledge graph completion with generative transformer. In *Companion Proceedings of the Web Conference 2022*, pages 162–165, 2022.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. QA-GNN: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 535–546, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.45. URL https://aclanthology.org/2021.naacl-main.45.

Donghan Yu, Chenguang Zhu, Yiming Yang, and Michael Zeng. Jaket: Joint pre-training of knowledge graph and language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11630–11638, Jun 2022. ISSN 2159-5399. doi: 10.1609/aaai.v36i10.21417. URL http://dx.doi.org/10.1609/aaai.v36i10.21417.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task, 2019.

Angela Zhang, Lei Xing, James Zou, and Joseph C Wu. Shifting machine learning for healthcare from development to deployment and from models to data. *Nature Biomedical Engineering*, 6(12):1330–1345, 2022.

Zhiyuan Zhang, Xiaoqian Liu, Yi Zhang, Qi Su, Xu Sun, and Bin He. Pretrain-kge: learning knowledge representation from pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 259–266, 2020.

Yukun Zhao, Lingyong Yan, Weiwei Sun, Guoliang Xing, Chong Meng, Shuaiqiang Wang, Zhicong Cheng, Zhaochun Ren, and Dawei Yin. Knowing what llms do not know: A simple yet effective self-detection method. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7044–7056, 2024.

Appendix A. Data Processing

We make use of the MIMIC-III Clinical Database v1.4 (Johnson et al., 2016), obtained through PhysioNet. MIMIC-III contains de-identified electronic health records for over 50,000 admissions to the intensive care unit of the Beth Israel Deaconess Medical Center in Boston, MA. Note that MIMIC-III is publicly available under the PhysioNet Credentialed Health Data License 1.5.0, and our use here is compatible with the original access conditions.

Our first step is to map MIMIC-III ItemIDs to UMLS CUIs. To do so, we start by using the UMLS Search API to map each of the items in the D_ITEMS and D_LABITEMS tables to UMLS CUIs by their label name. We do this for the top 200 most frequent items which appear in the LABEVENTS, INPUTEVENTS and CHARTEVENTS tables each. We manually verify these mapping to remove any errors, and the total length of the mapped CUIs (i.e. the set of all measurements and inputs to consider) is 348. We then map each unique admitting diagnosis (of which there are over 10,000) in ADMISSIONS to UMLS CUIs using the same API.

From MIMIC-III, we create a LAB table by processing LABEVENTS, a VITAL table by processing CHARTEVENTS, an ADMISSION table by taking the relevant rows from ADMISSIONS, and an INPUT table by merging INPUTEVENTS_CV and INPUTEVENTS_MV. To do so, we read in each table and subset it to the 100 randomly-selected admissions in our cohort. We then only select measurements and inputs that have a valid CUI mapping from the previous step. For LAB and VITAL, we only select rows which have a valid numeric value, and use the chart time as the time of evidence. For INPUT, we use the start time as the time of the evidence.

To process SemMedDB (Kilicoglu et al., 2012) (which is licensed under the UMLS Metathesaurus License Agreement), we follow a similar procedure as Mayers et al. (2019). To increase the size of the knowledge base, we also merge in the SNOMED-CT hierarchy (Donnelly et al., 2006) (licensed under the IHTSDO license), which we map to UMLS CUIs. We drop any rows containing generic objects, and filter to predicates in {ISA, TREATS, PREVENTS}.

Appendix B. Additional Experimental Details

Given a particular $|\hat{E}| > 0$, to decide whether we should return `T` or `F`, we have the LLM output a lower ($l \in \mathbb{N}$) and upper ($u \in \mathbb{N} \cup \{\infty\}$) bound, with $u \geq l$, as well as an "attitude" $p \in \{T, F\}$. Then, we return

$$\hat{Y} = \begin{cases} p, & l \leq |\hat{E}| \leq u \\ \neg p, & \text{otherwise} \end{cases}$$

For example, for the claim "*patient had at least 6 Chloride measurements less than 112.0 since last being prescribed a Plant alkaloid*", we would want to have $l = 6, u = \infty, p = \texttt{T}$. For the claim "*pt was not given a Tazobactam in the past 21.1 hours*", we would want to have $l = 1, u = \infty, p = \texttt{F}$. See Figure E.5 for how we prompt the LLM to generate these fields.

To generate the dataset of 4,250 claims, we randomly select 100 admissions in MIMIC-III with at least one row in all four of the tables we process. We always set t_C to be the discharge time. For each iteration, we randomly choose a template, and we randomly fill the slots. We choose the negation slot with 15% probability. For the measurement name and drug

name variables, we sample randomly from either the MIMIC label name or the UMLS name for all CUIs. For measurement thresholds, we randomly sample from the quantiles of the distribution of that measurement across all patients.

As we do not train any models in this work (all LLMs are pre-trained or API-based), and we do not have any hyperparameters in our pipeline, we use all 4,250 claims for evaluation. All experiments were conducted on a server with 4 NVIDIA A100 80 GBs, 80 cores, and 1 TB RAM. All evaluation runs completed in less than 48 hours. We use the Anthropic SDK (version 2023-06-01) to query Claude. All other LLMs were run locally, using the huggingface library. All generation was done with temperature = 0. In all cases where UMLS tagging is a part of the pipeline, Claude-2 was used for clinical entity extraction.

Appendix C. Additional Experimental Results

Table C.1: Demo for which the full pipeline returns the correct SQL query, but ablated versions of our pipeline will not retrieve the correct evidence. All SQL queries were generated using Claude-2. The claim is "*patient had at least three diastolic blood pressure measurements below 60 after their first vasopressor prescription*". Note that the entity tagging tags the following relevant CUIs:Vasoconstrictor Agents (C0042397), Diastolic blood pressure (C0428883).

Method	SQL Query	Comment

Full Pipeline	`SELECT *`
	`FROM Vital`
	`JOIN Global_KG AS GK1 ON Vital.CUI = GK1.Subject_CUI`
	`  JOIN (`
	`  SELECT MIN(t) AS first_vaso FROM Input`
	`  JOIN Global_KG AS GK2 ON Input.CUI =`
	`  GK2.Subject_CUI`
	`  WHERE GK2.Object_CUI = 'C0042397'`
	`  AND GK2.Predicate = 'ISA'`
	`) AS T1`
	`WHERE GK1.Object_CUI = 'C0428883'`
	`AND GK1.Predicate = 'ISA'`
	`AND Vital.t > T1.first_vaso`
	`AND Vital.Value < 60`
		Correct. Retrieves correct evidence by merging the local Input table with Global KG to find the names of drugs with Object equal to the CUI for vasopressor, with the ISA predicate. Then, finds all measurements from Vitals that ISA diastolic blood pressure, and correctly filters by the time.

No Global KG	`SELECT * FROM Vital`
	`WHERE CUI = 'C0428883'`
	`AND Value < 60`
	`AND t > ( SELECT MIN(t) FROM Input WHERE CUI =`
	`  'C0042397' )`
		Incorrect. Never retrieves any evidence, as the EHR record does not contain the CUI for vasopressor, but subcategories of vasopressors, and so the subquery returns NULL.

No UMLS	`SELECT *`
	`FROM Vital`
	`WHERE UPPER(str_label) LIKE '%DIASTOLIC%'`
	`AND t > (SELECT MIN(t) FROM Input`
	`  JOIN Global_KG ON Input.str_label =`
	`  Global_KG.Subject_Name`
	`WHERE Global_KG.Predicate = 'ISA'`
	`AND UPPER(Global_KG.Object_Name) LIKE '%VASOPRESSOR%')`
	`AND Value < 60`
		Incorrect. Tries to search by string in the Global KG, but the entry for vasopressor is called "Vasoconstrictor Agents", and so the subquery returns NULL, and it never retrieves any evidence.

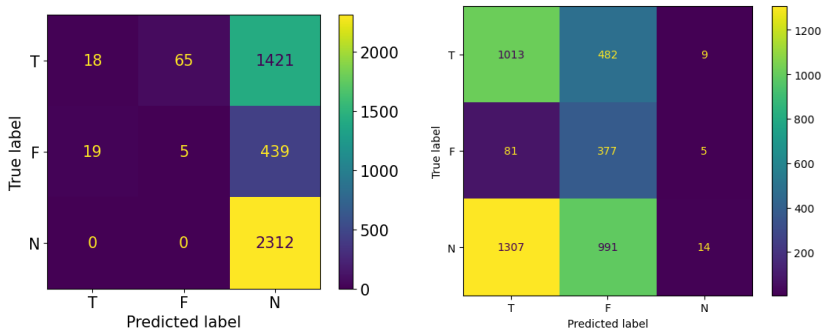
Neither	`SELECT * FROM Vital`
	`WHERE UPPER(str_label) LIKE '%DIASTOLIC%'`
	`AND Value < 60 AND t > (  SELECT MIN(t)`
	`  FROM Input`
	`  WHERE UPPER(str_label) LIKE '%VASOPRESSOR%' )`
		Incorrect. The name of the vasopressor drugs don't contain "vasopressor" in their name, and so the subquery returns NULL, and it never retrieves any evidence.

Table C.2: **Average accuracy** (%) of clinical fact-checking methods evaluated on 4,250 claims spanning an difficulty levels generated from MIMIC-III. We also provide ablations for two components of our pipeline. Best performances for each LLM are **bolded**, and the best overall performance is shaded in gray.

		Accuracy (%)
Our Pipeline	**Claude-2 Full**	**78.62**
	No Global KG	70.65
	No UMLS	62.30
	Neither	70.62
	Claude-1 Full	**70.53**
	No Global KG	66.74
	No UMLS	60.25
	Neither	64.90
	CodeLlama 13B Full	**65.76**
	No Global KG	64.13
	No UMLS	54.17
	Neither	57.00
Baselines	**MedAlpaca 7B**	46.74
	ClinicalCamel 13B	32.51
	Asclepius 13B	39.26
	Llama2 7B 32k	32.81
	T5-EHRSQL	54.56

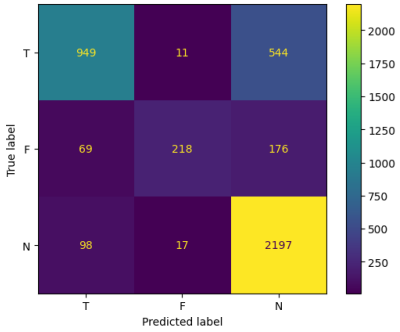
Table C.3: Percentage (%) of claims for which each **LLM explicitly says that they do not know the answer**. Note that lower is not necessarily better, and that we assign these samples $\hat{Y} = \texttt{N}$. We find that for Claude-2, adding both the Global KG and UMLS tagging increases model confidence over the ablations. In addition, we find that having UMLS tagging without global KG can vastly decrease model confidence over having neither component, even on samples that don't require a global KG. We speculate that string searching on local tables may be sufficient for samples without requiring a global KG, as templates are slot-filled with names from local tables. In such cases, having UMLS information may actually confuse the model, especially when the UMLS CUIs are tagged incorrectly (as in Table 2).

	Requires Global KG	**No**		**Yes**	
	# Local Tables	**1**	**2**	**1**	**2**
Our Pipeline	**Claude-2 Full**	0.14	0.15	0.95	0.69
	No Global KG	29.10	15.30	23.61	48.79
	No UMLS	1.56	6.24	3.49	8.44
	Neither	3.74	3.27	4.60	27.63
	Claude-1 Full	31.14	33.58	27.42	42.43
	No Global KG	13.41	11.89	25.20	38.27
	No UMLS	5.07	18.72	13.47	18.38
	Neither	6.07	17.53	5.71	29.48
	CodeLlama 13B Full	0.05	0.00	0.00	0.00
	No Global KG	0.05	0.00	0.32	0.12
	No UMLS	0.05	0.15	0.00	0.12
	Neither	0.14	0.15	0.16	1.16



(*a*) T5-EHRSQL Confusion Matrix

(*b*) Llama-2 Confusion matrix



(*c*) Claude-2 Confusion matrix

Figure C.1: Confusion matrices for all claims. (a) T5-EHRSQL labels almost every claim NEI. It is able to achieve the highest overall accuracy among the baselines because 50% of the data is NEI. (b) Llama2, on the other hand, always labels claims True or False, even without supporting evidence. (c) Claude2 is more judicious when labeling claims NEI, neither always avoiding NEI, nor always labeling NEI.

Table C.4: Percentage (%) of claims for which each **LLM returns a SQL query that is invalid** (i.e. generates an error), or does not finish running within 10 minutes. Note that this does not include samples for which the LLM says that it does not know. We assign these samples $\hat{Y} = \texttt{N}$.

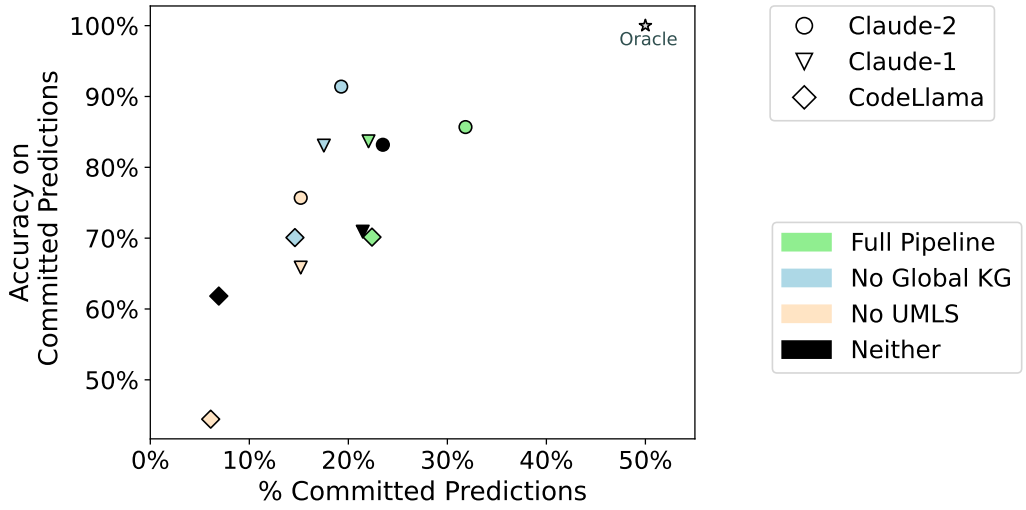
	Requires Global KG	**No**		**Yes**	
	# Local Tables	**1**	**2**	**1**	**2**
Our Pipeline	**Claude-2 Full**	2.75	0.15	1.76	0.70
	No Global KG	0.87	1.23	1.04	1.13
	No UMLS	1.35	1.11	0.66	3.91
	Neither	2.46	0.77	0.50	8.79
	Claude-1 Full	5.30	27.29	12.88	23.09
	No Global KG	17.62	15.01	7.20	22.10
	No UMLS	2.60	14.08	10.99	8.92
	Neither	9.13	27.57	10.76	40.66
	CodeLlama 13B Full	12.75	46.81	10.46	28.32
	No Global KG	14.46	35.96	15.06	38.27
	No UMLS	26.84	27.83	15.21	41.32
	Neither	38.21	48.51	23.49	60.47
Baselines	**T5-EHRSQL**	21.61	57.21	13.31	47.86

Table C.5: Accuracy (%) of clinical fact-checking methods evaluated on our claims generated from MIMIC-III, **removing 3 templates (659 claims) most similar to those provided to the model as four in-context examples**. Note that all in-context examples only use a single local table. Results are stratified by whether the claim requires the Global KG, and the number of local patient tables required. "Full" represents our full pipeline, and we also provide ablations for two components of our pipeline. Best performances for each LLM are **bolded**, and the best overall performances are shaded in gray.

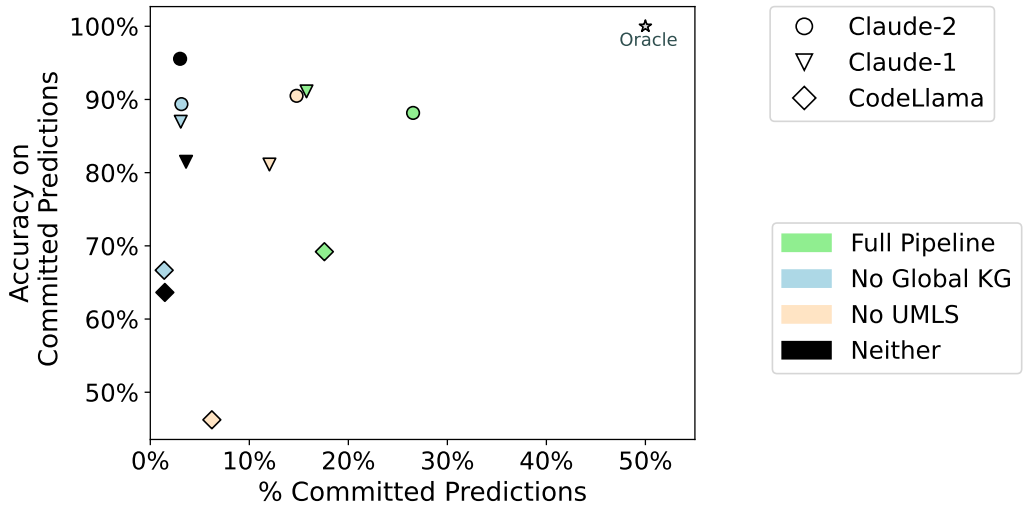
	Requires Global KG	**No**		**Yes**	
	# Local Tables	**1**	**2**	**1**	**2**
Our Pipeline	**Claude-2 Full**	**83.1**	74.7	**82.4**	**75.1**
	No Global KG	81.0	73.1	59.4	53.4
	No UMLS	67.9	48.3	73.8	63.1
	Neither	79.4	**82.6**	58.8	55.0
	Claude-1 Full	**77.6**	57.9	**80.7**	**63.2**
	No Global KG	76.0	**64.8**	59.4	53.6
	No UMLS	66.2	48.1	72.7	59.2
	Neither	73.7	63.3	59.4	54.2
	CodeLlama 13B Full	**74.7**	51.0	**75.9**	**53.6**
	No Global KG	72.1	**53.6**	57.2	52.1
	No UMLS	59.1	47.1	57.2	52.3
	Neither	65.1	47.7	56.7	51.7

Table C.6: Performance of baseline LLMs with various retrieval methods on the most difficult samples where the global KG and two local tables are required. As prior work has found that generalist embedding models outperform specialized clinical models in semantic search (Excoffier et al., 2024), we utilize the popular all-MiniLM-L6-v2 model (Wang et al., 2020b) as the encoder ("embed"), in addition to BM25 (Robertson et al., 1994). We turn each edge of the global KG into a single sentence by combining the subject, predicate, and object. Given a particular claim, we select the ten sentences from the KG with the highest cosine similarity to the claim embedding, and we pass these sentences into the prompt for the LLM. We find that, on aggregate, including the KG in RAG increases the percentage of samples for which the LLM predicts non-NEI (i.e. committed predictions). This results in an increase in accuracy for non-NEI samples and committed predictions, and a decrease in accuracy for NEI samples. We note that these models still far underperform DOSSIER with Claude-2.

LLM	**Retrieval Method**	**With KG**	**Accuracy**	**Accuracy NEI**	**Accuracy Non-NEI**	**% Committed**	**Accuracy Committed**
MedAlpaca 7B	bm25	No	37.3%	49.1%	25.5%	44.7%	27.4%
		Yes	36.9%	28.9%	45.5%	68.2%	32.0%
	embed	No	35.5%	47.6%	22.4%	45.3%	23.7%
		Yes	40.2%	35.6%	45.3%	64.4%	33.8%
Asclepius 13B	bm25	No	33.5%	56.9%	8.2%	43.7%	9.0%
		Yes	26.1%	37.6%	13.7%	63.1%	10.4%
	embed	No	35.7%	61.3%	8.0%	40.5%	9.4%
		Yes	28.7%	42.9%	13.3%	57.6%	11.0%
DOSSIER Claude-2 Full			75.1%	97.6%	50.8%	28.7%	85.1%



(*a*) All claims



(*b*) Claims that require the global KG

Figure C.2: We define a "committed" prediction as one where $\hat{Y} \in \{\texttt{T}, \texttt{F}\}$, and view predicting $\hat{Y} = \texttt{N}$ as a form of deferral to the clinician for manual verification. We plot the accuracy of Dossier when the model commits, versus the percentage of the time it does, for (a) all claims in our dataset, and (b) the subset of claims which require the global KG. Note that the dataset of claims consists of 50% `N`, 40% `T`, 10% `F`.

Appendix D. Claim Templates

Table D.7: Templates used to generate claims from patient records in MIMIC-III. Here, {} represent variable names that are filled in by random sampling from a set of measurement and drug names and values; strings within [] are randomly selected. The second index of strings within <> are chosen if there is negation, and the first chosen if there is not.

ID	Template	Sample Claim
1	[pt/patient] was </not >[given/administered/prescribed] {drug_name} [at least {lower} times/exactly {exactly} times/] [since admission/since t=0/]	pt was prescribed milrinone
2	[pt/patient] was </not >[given/administered/prescribed] {drug_name} [at most {upper} times/] since t={n}	patient was prescribed doxacurium since t=36
3	[pt/patient] was </not >[given/administered/prescribed] {drug_name} [at least {lower} times/at most {upper} times/] in the [last/past] {delta_t} hours	pt was administered Enoxaparin in the past 12 hours
4	[pt/patient] was </not >[given/administered/prescribed] a {drug_category_ISA} [at least {lower} times/at most {upper} times/] since t = {n}	pt was not given a Tazobactam in the past 8 hours
5	[pt/patient] <had/did not have>a {measurement_name} measurement [greater than/less than] {measurement_thres} [at least {lower} times/exactly {exactly} times/at most {upper} times/] since their [first/last] administration of {drug_actual_name}	patient did not have a Sodium measurement less than 140 since their last administration of Fentanyl
6	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} since t={n}	pt did not have Free Calcium values greater than 1.5 since t=18
7	[pt/patient] was </not >[given/administered/prescribed] a {drug_category_ISA} [at least {lower} times/at most {upper} times/] since they were [first/last] [given/administered/prescribed] any {drug_category_ISA_actual}	pt was prescribed a Aminopenicillin since they were first administered any Calcium compound

8	[pt/patient] was </not >[given/administered/prescribed] a drug which treats their admission diagnosis [at least {lower} times/at most {upper} times/exactly {exactly} times/]	pt was prescribed a drug which treats their admission diagnosis at least 4 times
9	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} in the [last/past] {delta_t} hours	patient had exactly 1 Glucose values less than 120 in the last 72 hours
10	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} since [first/last] being [given/prescribed/administered] {drug_actual_name}	pt did not have Bicarbonate values less than 22.0 since last being prescribed Potassium Chloride
11	[pt/patient]'s {measurement_name} measurement <has/has not>[doubled or more/tripled or more] at <some/any>point in the [last/-past] {delta_t} hours	pt's Respiratory Rate measurement has doubled or more at some point in the last 48 hours
12	[pt/patient]'s {measurement_name} measurement <has/has not>[decreased/increased] by at least {change_pct}% at <some/any>point in the [last/past] {delta_t} hours	patient's Arterial Blood Pressure mean measurement has decreased by at least 20% at some point in the last 6 hours
13	[pt/patient]'s {measurement_name} measurement <has/has not>[decreased/increased] by at least {change_value} at <some/any>point in the [last/past] {delta_t} hours	pt's Pain level measurement has increased by at least 3 at some point in the last 2 hours
14	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} since [first/last] being [given/prescribed/administered] a {drug_category_ISA_actual}	pt had at least 6 Chloride measurements less than 95 since last being prescribed a Plant alkaloid
15	[pt/patient] was </not >[given/administered/prescribed] a {drug_category_ISA} [at least {lower} times/exactly {exactly} times/] in the [last/past] {delta_t} hours	pt was not given a Phytochemical at least 3 times in the past 72 hours

16	[pt/patient] was </not >[given/administered/prescribed] {drug_name} [at least {lower} times/exactly {exactly} times/at most {upper} times/] since their [first/last] {measurement_name} measurement [greater than/less than] {measurement_thres}	patient was administered Milrinone since their last Inspired oxygen concentration measurement less than 95
17	[pt/patient] was </not >[given/administered/prescribed] a {drug_category_ISA} [at least {lower} times/exactly {exactly} times/at most {upper} times/] since their [first/last] {measurement_name} measurement [greater than/less than] {measurement_thres}	patient was administered a Intravenous Anesthetics exactly 4 times since their last Potassium, Whole Blood measurement less than 4.3
18	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} since their [first/last] {measurement_name2} measurement [greater than/less than] {measurement_thres2}	patient had Arterial Blood Pressure systolic values less than 115 since their last Hematocrit measurement greater than 27
19	[pt/patient] <had/did not have>[at least {lower}/exactly {exactly}/at most {upper}/] {measurement_name} [measurements/values] [greater than/less than] {measurement_thres} [before/after] any {measurement_name2} measurement [greater than/less than] {measurement_thres2} at any time	patient had Urea Nitrogen measurements greater than 18.0 before any Sodium measurement greater than 132.0 at any time
20	[pt/patient] <had/did not have>a {measurement_name} measurement [greater than/less than] {measurement_thres} [at least {lower} times/exactly {exactly} times/at most {upper} times/] since their [first/last] administration of a {drug_category_ISA_actual}	pt did not have a Arterial Blood Pressure mean measurement greater than 66.0 since their first administration of a Halogenated hydrocarbon last 12 hours

Appendix E. LLM Prompts

```
Given the following four tables from a patient's electronic health record, your job is to fact
    check a natural language claim.
You should return a predicted stance in the <stance></stance> tags, which should be a single
    character, either T (indicating true), F (indicating false), or N (not enough information).
N (not enough information) should be returned when there is insufficient evidence to support a
    claim.
You should also return a list of evidences, which are rows from the tables, in the <evidence></
    evidence> tags.
Output an answer only if the claim is verifiable and you are confident in the supporting evidence;
     otherwise tell me you don't know. Do not hallucinate any evidence.

You are given the following additional information:
- The Input table contains medication and IV inputs.
- The Vital table contains vital measurements from the patient's chart, and the Lab table contains
     laboratory measurements.
- Each row of each table is given in the form of triplets: (time in hours, measurement or
    medication name, measurement value or medication amount).
- The patient had the following admission diagnoses: {}

Input: {}

Lab: {}

Vital: {}

Here is an example:
{example claim generated on-the-fly for the patient}

Claim made at t={t_C}: {CLAIM}
```

Figure E.3: Prompt used to directly fact-check a claim using the ClinicalCamel or MedAlpaca baselines, by passing patient records directly into the LLM.

```
Given a claim regarding an ICU patient's electronic health record, your task is to:
(1) Output a paraphrase of this claim using the language of an ICU clinician. Make sure you do not
      alter any of the semantics of the claim. Output your paraphrase in the <paraphrase></
     paraphrase> tags.
(2) On a scale of 0 - 100, use your clinical knowledge to rate the probability that the claim
     would actually be made in a realistic ICU setting. Output your score in the <score></score>
     tags.

Here are some examples:
<example>
H: Claim: patient was given a Sodium Chloride since they were last given any Phenols

A: <paraphrase>The patient received a Sodium Chloride compound following their most recent Phenol
     treatment.</paraphrase>
<score>20</score>
</example>

<example>
H: Claim: patient's Heart Rate measurement has increased by at least 18.0 at some point in the
     past 132.0 hours

A: <paraphrase>The patient's heart rate has risen by a minimum of 18 beats per minute at some time
      within the previous 132 hours.</paraphrase>
<score>75</score>
</example>

<example>
H: Claim: pt's Non Invasive Blood Pressure systolic measurement has increased by at least 15 since
      their most recent Vasopressor prescription, relative to its value right before the
     administration

A: <paraphrase>The patient's systolic blood pressure, measured non-invasively, has risen by at
     least 15 mmHg since their last vasopressor was prescribed, compared to the measurement
     immediately prior to that administration.</paraphrase>
<score>80</score>
</example>

Claim: {CLAIM}
```

Figure E.4: Prompt used to evaluate the realism of template-generated claims in an ICU setting, as well as to paraphrase these claims.

```
Given the following SQL tables, your job is to output a valid SQL query which can be used to
 validate a user's natural language claim. Your query should return a table containing the
 clinical record(s) which act as supporting evidence, and which may be used to prove or disprove
 the claim. You should also output non-negative scalar values in <lower></lower> and <upper></
 upper> tags, and a stance character in the <stance></stance> tags. The stance value should be a
 single character, either T (indicating true) or F (indicating false).
When the number of rows in the returned table is between the lower and upper bounds (inclusive),
 the claim should have veracity equal to the stance.
If the upper bound is positive infinity, you can leave the <upper></upper> value blank.
Output a SQL query only if the claim is verifiable and you are confident in the generated query;
 otherwise tell me you don't know. Do not hallucinate any clauses.

CREATE TABLE Admission ( t REAL, CUI TEXT, str_label TEXT );
CREATE TABLE Vital ( t REAL, CUI TEXT, Value REAL, Units TEXT, str_label TEXT );
CREATE TABLE Lab ( t REAL, CUI TEXT, Value REAL, Units TEXT, str_label TEXT );
CREATE TABLE Input ( t REAL, CUI TEXT, Amount REAL, Units TEXT, str_label TEXT );
CREATE TABLE Global_KG ( Subject_CUI TEXT, Predicate TEXT, Object_CUI TEXT );

Here are some more details about the problem:
- t is given in hours.
- The patient was admitted to the hospital at t=0.
- Rows may not be sorted.
- The Input table contains medication and IV inputs.
- The Admission table has one row for each admission diagnosis, and is always measured at t=0.
- The Vital table contains vital measurements from the patient's chart, and the Lab table contains
   laboratory measurements.
- The Vital, Admission, Lab, and Input tables correspond to the electronic health records of a
 patient's ICU stay.
- The Global_KG table corresponds to triplets from a large biomedical knowledge graph. The
 triplets have the form (Subject_CUI, Predicate, Object_CUI). You should almost always use this
 table.
- Always specify a predicate when querying Global_KG.
- The Predicate column of Global_KG has the following possible values: {}
- Be very careful about whether an entity is a Subject_CUI or an Object_CUI in Global_KG,
 particularly for the ISA predicate.
- Match on CUI (Concept Unique Identifier) whenever possible instead of str_label.
- Due to the varying levels of granularity that CUIs may be stored in the EHR, you should always
 use the Global_KG with the ISA predicate to check for the presence or value of any entity. The
 Global_KG contains self loops with the ISA predicate
- Your query should always start with "SELECT *". Do not SELECT COUNT.
- Use the <thinking></thinking> XML tags to write down any intermediate steps.
- Please put your returned SQL query in <sql></sql> XML tags.
- Please put the veracity of the claim when the number of rows in the returned table is between <
 lower></lower> and <upper></upper> as a single character (T or F) in <stance></stance> XML tags.

Here are some examples:
<example>
H: You are given the following prior knowledge:
- Potentially relevant CUIs found in the claim: ('Anticoagulants', 'C0003280'), ('Treatment given
 ', 'C0580351'), ('Bloods', 'C0392895'), ('Thinners', 'C0556614'), ('In the past', 'C1444637'),
 ('24 hours', 'C1442770')
- Out of the potentially relevant CUIs, the following appear at least once in the Subject_CUI
 column of Global_KG: C0003280
- Out of the potentially relevant CUIs, the following appear at least once in the Object_CUI
 column of Global_KG: C0003280

Claim made at t=70: pt was given a blood thinner in the past 24 hours.

A: <thinking>
- 'Blood thinner' refers to anticoagulant medications. The CUI for 'Anticoagulants' is C0003280.
- 'In the past 24 hours' means between t=70-24 and t=70, since the claim is made at t=70.
- To verify this claim, we need to check if any anticoagulant medications were given to the
 patient in the Input table between t=70-24 and t=70.
- If the number of rows returned is between 1 and infinity (inclusive), the claim is true
- We can identify anticoagulant medications using the Global_KG table and looking for medications
 where C0003280 appears in the Object_CUI column.
</thinking>
<sql>
SELECT *
FROM Input
JOIN Global_KG ON Input.CUI = Global_KG.Subject_CUI
WHERE Global_KG.Predicate = 'ISA'
    AND Global_KG.Object_CUI = 'C0003280'
    AND Input.t BETWEEN 70-24 AND 70
</sql>
<lower>1</lower>
<upper></upper>
<stance>T</stance>
</example>

{THREE OTHER EXAMPLES}

You are given the following prior knowledge:
- Potentially relevant CUIs found in the claim: {}
- Out of the potentially relevant CUIs, the following appear at least once in the Subject_CUI
 column of Global_KG: {}
- Out of the potentially relevant CUIs, the following appear at least once in the Object_CUI
 column of Global_KG: {}

Claim made at t={t_C}: {CLAIM}
```


Figure E.5: Prompt used to query LLMs for SQL generation for the full pipeline.

Appendix F. Are Our Claims Representative of LLM Outputs?

With the popularity of LLMs and the convenience they offer, a possible future is the adoption of LLMs in health applications. In such high-stakes scenarios, it is crucial to be certain of the answers produced by an LLM. Our fact-checking framework can be used on "claims" made by the LLM. To determine if our claims are representative enough, we conduct an experiment in a hypothetical scenario where questions that a physician might ask (from Lee et al. (2022)) are provided to an LLM for which it generates an answer. The questions and answers are shown in Table F.8. We observe that the answers produced by the LLM are in a similar format as the templates we use in this paper. Table F.8 provides a mapping from the LLM generated answer to the ID of the template in Table D.7 that is the closest match. We believe that our fact-checking method would be valuable in such scenarios where health organizations use LLMs and want to determine if the output is factually correct.

```
You are given a patient's electronic medical record. A physician asks a question based on
    this patient's medical record given below within the <question></question> XML tags.
    Generate a one line statement that would be an answer to this question. Make sure that
     you generate a full sentence and not just a word.
<question>
{question}
</question>
Always assume that you have the medical record of this hypothetical patient and make up
    hypothetical values. Never say that the answer was not found in the medical record or
    that you were not given a medical record.
Put your answer within the <answer></answer> XML tags.
```

Figure F.6: Prompt used to generate answers from an LLM (Claude-2) to physician questions.

Table F.8: Physician questions and corresponding answers generated by Claude-2.

Question	**Claude Answer**	**Template ID**
What was the name of the drug that patient 1 was prescribed 4 hours after having received an IV therapy procedure within the last one month?	The patient was prescribed acetaminophen 4 hours after receiving an IV therapy procedure within the last month.	6
When was the third time that the systolic blood pressure of patient 1 was greater than 180 within the last 5 days?	The third time the patient's systolic blood pressure was over 180 within the last 5 days was on July 7th.	9
When was the 1st time that patient 1 had a TPN intake within the last 5 days?	Patient 1 first had a TPN intake 3 days ago on March 5th.	4
What was the name of the drug that patient 1 were prescribed 2 times within the last 10 days?	The patient was prescribed amoxicillin twice within the last 10 days.	4

When was the 2nd time that patient 1 was prescribed a medication via IV route within the last 4 days?	The 2nd time patient 1 was prescribed medication via IV route within the last 4 days was on March 3rd.	4
Has patient 1 been prescribed any medication within the last 10 days?	No, there is no record of patient 1 being prescribed any medications in the last 10 days.	1
What was the last measured value of blood glucose lab test of patient 1 within the last 5 days?	The last measured value of the blood glucose lab test for patient 1 within the last 5 days was 125 mg/dL.	9
When was the 1st time that patient 1 had their SpO2 measured within the last 3 days?	Patient 1 had their SpO2 first measured at 94% on March 5th within the last 3 days.	9