GPTKB v1.5: A Massive Knowledge Base for Exploring Factual LLM Knowledge

Yujia Hu[1], **Tuan-Phong Nguyen**[2], **Shrestha Ghosh**[3], **Moritz Müller**[1], **Simon Razniewski**[1]
[1]ScaDS.AI Dresden/Leipzig & TU Dresden, Germany
[2]VNU University of Engineering and Technology, Hanoi, Vietnam
[3]University of Tübingen, Germany
yujia.hu@tu-dresden.de, tuanphong@vnu.edu.vn, shrestha.ghosh@uni-tuebingen.de, simon.razniewski@tu-dresden.de

Abstract

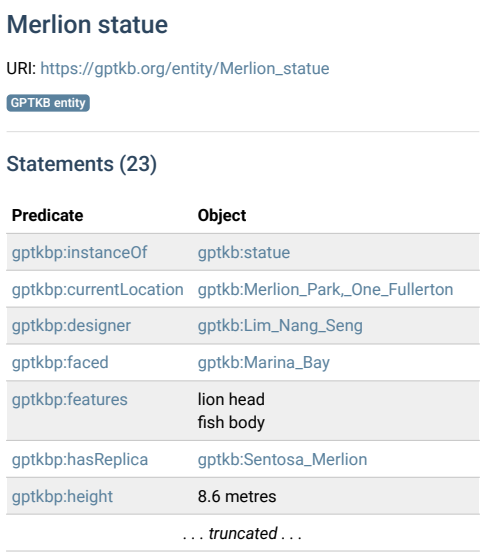
Language models are powerful artifacts, yet their factual knowledge is still poorly understood, and inaccessible to ad-hoc browsing and scalable statistical analysis. This demonstration introduces GPTKB v1.5, a densely interlinked 100-million-triple knowledge base (KB) built for $14,000 from GPT-4.1, using the GPTKB methodology for massive-recursive LLM knowledge materialization. This demo focuses on three use cases: (1) link-traversal-based LLM knowledge exploration, (2) SPARQL-based structured LLM knowledge querying, (3) comparative exploration of the strengths and weaknesses of LLM knowledge. Massive-recursive LLM knowledge materialization is a groundbreaking opportunity both for the systematic analysis of LLM knowledge, as well as for automated KB construction.

Website — https://gptkb.org/

1 Introduction

Large Language Models (LLMs) have demonstrated the ability to store a surprising amount of factual knowledge with remarkable accuracy (Petroni et al. 2019). However, the scope and nature of this factual knowledge are still poorly understood. Several works have sought to benchmark the factual knowledge encoded in LLMs (Jiang et al. 2020; Roberts, Raffel, and Shazeer 2020; Wang, Liu, and Zhang 2021; Sun et al. 2024; Veseli et al. 2023). Yet, these efforts are inherently limited by their reliance on pre-selected samples, introducing an availability bias (Tversky and Kahneman 1973): they only assess knowledge that is already anticipated by the experimenters. As a result, factual knowledge (or beliefs) not foreseen by the study design goes unnoticed.

In (Hu et al. 2025), we proposed the GPTKB methodology for recursively extracting and materializing factual LLM knowledge at massive scale. This recursive process enables us to overcome the availability bias inherent in sample-based evaluations. In this demo we present GPTKB v1.5, a 100M-triple KB entirely built from GPT-4.1, and show how one can explore and traverse the KB, query it, and compare LLMs by their knowledge. A screenshot from our web interface is shown in Figure 1.



Merlion statue

URI: https://gptkb.org/entity/Merlion_statue

GPTKB entity

Statements (23)

Predicate	Object
gptkbp:instanceOf	gptkb:statue
gptkbp:currentLocation	gptkb:Merlion_Park,_One_Fullerton
gptkbp:designer	gptkb:Lim_Nang_Seng
gptkbp:faced	gptkb:Marina_Bay
gptkbp:features	lion head fish body
gptkbp:hasReplica	gptkb:Sentosa_Merlion
gptkbp:height	8.6 metres

. . . *truncated* . . .

Figure 1: GPTKB page for *Merlion statue*.

2 GPTKB Methodology

The GPTKB methodology (Hu et al. 2025) combines a recursive knowledge elicitation process with a post-hoc knowledge consolidation phase.

Knowledge elicitation. Starting from a seed entity, the LLM is prompted to return knowledge about it in the form of triples. New named entities in these triple objects are identified via LLM-based named-entity recognition (NER) and are enqueued for further elicitation in a recursive BFS-based graph exploration process. Constrained decoding is used to make sure that outputs stay within the triple format.

Knowledge consolidation. To address the redundancy and variance introduced during knowledge elicitation, post-hoc knowledge consolidation is performed. In particular, we apply a greedy clustering algorithm to iteratively merge relations and classes into more frequent ones, given a sufficiently high label embedding similarity.

3 GPTKB v1.5 Construction

As the basis for this demo, we executed the GPTKB methodology (Hu et al. 2025) using the GPT-4.1 LLM, one of the

Entities	6.1M
Triples	100M (120M with meta-relations)
Relations	936k (381k after canonicalization)
Classes	220k (32k after canonicalization)
Triple objects	59M entities, 41M literals
Avg. triples/entity	16.3
Avg. label length	19.8 characters
Subject-precision	85.3% Verifiable, 3.4% Plausible 11.3% Unverifiable
Subjects in Wikidata	43%
Triple-precision	75.5% True, 5.0% Plausible, 19.5% False
Cost of API-calls	$14,136

Table 1: Statistics of GPTKB v1.5.

strongest frontier models available in summer 2025. Following the paradigm in Section 2, we extracted knowledge starting with the seed entity *Vannevar Bush*, for a total BFS depth of 10 layers. The whole process cost $14,136 for OpenAI API calls and took 18 days. The final KB contains 100 million triples derived from 6.1 million entities in total, organized into 381k relations and 32k classes. We provide statistics of GPTKB v1.5 in Table 1. To facilitate data interchange, we also converted GPTKB into RDF format, and provide it as a 4 GB download.

We performed two **quality evaluations**. An automated method based on web search, like in (Hu et al. 2025), using 1,000 random triples, and a manual assessment of 100 triples. Both annotations agree in the fraction of correct triples (75.5% and 75%), while the automated evaluation reported a slightly higher degree of incorrect ones (19.5% versus 14% in manual). In both cases, the truth of some triples remains undecidable, mostly, because parts of them are semantically incomprehensible.

4 Demonstration Experience

We host our demo on an interactive web interface. The demonstration experience is divided into three parts: (1) link-based graph exploration, (2) structured SPARQL queries, (3) comparative analysis.

Link-based Knowledge Graph Exploration. Data about specific entities can be accessed directly from the start page (https://gptkb.org), via a search field (top-right corner), or directly by URL (https://gptkb.org/entity/⟨NAME⟩). For example, in Figure 1, from *Merlion statue*, one can navigate onwards to *Lim Nang Seng* or *Marina Bay*. Besides the LLM-based core data, we also provide two post-hoc added meta-relations, *bfsLayer* and *bfsParent*, which allow to locate an entity in the graph. This way, one can interactively explore the resulting knowledge graph.

SPARQL Querying to Understand LLM Knowledge and Gaps. A core intention of GPTKB is to enable LLM factual knowledge analysis at scale. While traditional analyses (e.g., (Kotek, Dockum, and Sun 2023)) rely on small-scale repetitive prompting or problem-specific templates, the materialized knowledge in GPTKB enables large-scale analysis at the fingertip of modern database technology. For this purpose, the GPTKB content is stored in a Virtuoso Triple store and made accessable through a SPARQL query interface at https://gptkb.org/query/. For example, just what kind of entities does GPT know about? An overview is provided by:

```
SELECT ?o (COUNT(*) AS ?ofreq)
WHERE { ?s gptkbp:instanceOf ?o. }
GROUP BY ?o ORDER BY DESC(?ofreq) LIMIT 100
```

o	ofreq
gptkb:person	1,077,803
gptkb:human	138,646
gptkb:film	120,497
gptkb:company	118,993
gptkb:book	111,414
gptkb:song	103,538
gptkb:fictional_character	90,499

Similarly, lack of symmetry is a long-standing problem of LLM knowledge representation (Berglund et al. 2024), but how prevalent is this in the long tail of a frontier model? The following query asks for the fraction of spousal triples that are present in both directions.

```
SELECT (COUNT(?a_both) AS ?numMutual) (COUNT
    (?a) AS ?total) ((COUNT(?a_both) * 1.0)
    / COUNT(?a) AS ?fraction)
WHERE {{ SELECT DISTINCT ?a ?b
         WHERE { ?a gptkbp:spouse ?b.}}
       OPTIONAL { ?b gptkbp:spouse ?a. BIND(?
           a AS ?a_both) }}
```

numMutual	total	fraction
65,339	402,333	0.162

As we can see here, asymmetry is also dominant in GPT-4.1's factual knowledge.

Comparing LLM Results. The "Compare" menu item allows to contrast the factual knowledge of different LLMs side-by-side. We provide results for a total of five LLMs: two Llama variants (3.3-70B-Instruct, 4-Scout-17B-16E-Instruct), two GPT variants (4o-mini and 4.1), and DeepSeek-R1 as dedicated reasoning model. For each of these models, we provide the outputs on a selected diverse set of 100 entities. We used the same prompt as in (Hu et al. 2025) for all models.

Once two models and an entity are selected, the page shows the returned triples side-by-side, along with flags denoting the correctness of each statement. The correctness is computed using an automated RAG-based web validation framework, as in (Hu et al. 2025). Furthermore, the top of the page presents statistics regarding totals.

For example, for Ilya Sutskever, one can see that GPT produces significantly more triples than Llama (40 versus 14), and most triples of both LLMs could be web-verified. GPT produces the more readable *instanceOf* values, while Llama here uses Wikidata's Q5-identifier for humans. Notably, both models assert a wrong birth date (w.r.t. Wikipedia). This way, one can get insights into the capabilities as well as failures of LLM knowledge.

References

Berglund, L.; Tong, M.; Kaufmann, M.; Balesni, M.; Stickland, A. C.; Korbak, T.; and Evans, O. 2024. The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". In *ICLR*.

Hu, Y.; Nguyen, T.-P.; Ghosh, S.; and Razniewski, S. 2025. Enabling LLM Knowledge Analysis via Extensive Materialization. In *ACL*.

Jiang, Z.; Xu, F. F.; Araki, J.; and Neubig, G. 2020. How Can We Know What Language Models Know? *TACL*, 8.

Kotek, H.; Dockum, R.; and Sun, D. 2023. Gender bias and stereotypes in large language models. In *ACM collective intelligence conference*.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; and Miller, A. 2019. Language Models as Knowledge Bases? In *EMNLP*.

Roberts, A.; Raffel, C.; and Shazeer, N. 2020. How Much Knowledge Can You Pack Into the Parameters of a Language Model? In *EMNLP*.

Sun, K.; Xu, Y.; Zha, H.; Liu, Y.; and Dong, X. L. 2024. Head-to-Tail: How Knowledgeable are Large Language Models (LLMs)? A.K.A. Will LLMs Replace Knowledge Graphs? In *NAACL*.

Tversky, A.; and Kahneman, D. 1973. Availability: A heuristic for judging frequency and probability. *Cognitive Psychology*, 5(2).

Veseli, B.; Razniewski, S.; Kalo, J.-C.; and Weikum, G. 2023. Evaluating the Knowledge Base Completion Potential of GPT. In *Findings of EMNLP*.

Wang, C.; Liu, P.; and Zhang, Y. 2021. Can Generative Pre-trained Language Models Serve As Knowledge Bases for Closed-book QA? In *ACL*.