Introducing GPTKB to the Semantic Web

Yujia Hu[1], Tuan-Phong Nguyen[2], Shrestha Ghosh[3], Moritz Müller[1] and Simon Razniewski[1]

[1]ScaDS.AI Dresden/Leipzig & TU Dresden, Germany
[2]Institute for AI, VNU University of Engineering and Technology, Hanoi, Vietnam
[3]University of Tübingen, Germany

Abstract

Knowledge bases (KBs) are a cornerstone of the Semantic Web, yet they still struggle with scale and scope, and their construction and curation still involve a lot of manual effort. Large language models (LLMs) have recently emerged as powerful tools for a range of tasks, yet their potential for automated KB construction is still poorly understood.

In this demonstrator, we showcase GPTKB, a methodology and KB entirely built from GPT-4.1. GPTKB is constructed by massive-recursive LLM knowledge materialization [1], using over 9M API calls for $14,000 to construct a 100M-triple knowledge base with over 6M entities.

Our demonstration focuses on two use cases: (i) Link-based KG exploration and (ii) SPARQL-based analysis and comparison to Wikidata. The GPTKB demonstrator is accessible at https://gptkb.org.

1. Introduction

Knowledge bases (KBs) like Wikidata [2], Yago [3] and DBpedia [4] are a cornerstone of the Semantic Web. Despite years of research efforts, public knowledge bases are still scarce, and limited in one way or another by scale, scope, timeliness, or quality. The dominant data source for public KBs remain human curators (Wikidata) and data integration (Yago, Wikidata) or semi-structured scraping (DBpedia), with alternative paradigms based on text extraction (ReVerb [5], NELL [6]) not achieving comparable success.

Recently, large language models (LLMs) emerged as powerful tools for a range of tasks, and their potential is also debated in the Semantic Web community [7]. In [1], we introduced the GPTKB methodology for massive-recursive knowledge materialization from LLMs. The present demonstration showcases the KB resulting from this work, GPTKB, in an interactive KB browser that includes a SPARQL query interface.

In particular, we showcase GPTKB v1.5, a 100M triple knowledge base extracted from GPT-4.1 using over 9M API calls, at a cost of $14,000. GPTKB v1.5 provides a unique view of the potentials offered, as well as the challenges faced by LLM-based KB constructions. We focus on two use cases:

1. Link-based interactive knowledge graph (KG) exploration;
2. SPARQL-based analysis and comparison to Wikidata.

Table 1 gives basic KB statistics, Table 2 provides a comparison to other prominent KBs.

2. GPTKB Methodology

The GPTKB methodology [1] combines a recursive knowledge elicitation process with a post-hoc knowledge consolidation phase.

Knowledge elicitation Starting from a seed subject, 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

ISWC 2025 Companion Volume, November 2–6, 2025, Nara, Japan
yujia.hu@tu-dresden.de (Y. Hu); tuanphong@vnu.edu.vn (T. Nguyen); shrestha.ghosh@uni-tuebingen.de (S. Ghosh); simon.razniewski@tu-dresden.de (S. Razniewski)

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.

KB	#entities	#assertions
Wikimedia-related		
Wikidata	113M	1.62B
Wikidata5m	5M	20M
Yago 4.5	50M	140M
DBpedia	3.8M	75M
Text-extracted		
NELL	?	12M
ReVerb	?	15M
Generative		
GPTKB v1.5	6.1M	100M

Table 2
Size comparison of major KBs. Sources in [1].

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.

Further methodological details can be found in Hu et al. [1].

3. GPTKB Construction

Two versions of GPTKB are available, GPTKB v1.1, based on GPT-4o-mini [1], and GPTKB v1.5, based on GPT-4.1 [8]. While GPTKB v1.1 was the first proof of the viability of our methodology, the output quality achieved was below expectations. In particular, more than 60% of the triples were estimated to be hallucinations, and significant problems occurred with output skew, with some entities having over 100k (virtually entirely hallucinated) triples. For v1.5, we therefore decided to use a significantly stronger LLM. We opted for GPT-4.1, because it is among the strongest frontier models available as of Summer 2025, and released less than 3 months ago.

Following the paradigm described in Section 2, We extracted knowledge from GPT-4.1 starting with the seed entity *Vannevar Bush*. 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.

Since crawl parallelization distorts BFS search order, we post-hoc recomputed the shortest paths of each node to the root, and stored this information in two meta-relations, *bfsLayer* and *bfsParent*, to enable structural insights. To facilitate data interchange, we also converted GPTKB into RDF format, and serialized it into Turtle syntax.

We performed two **quality evaluations**. An automated method based on web search, like in Hu et al. [1], 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. GPTKB Web Provision

We provide multiple modes of access to GPTKB.

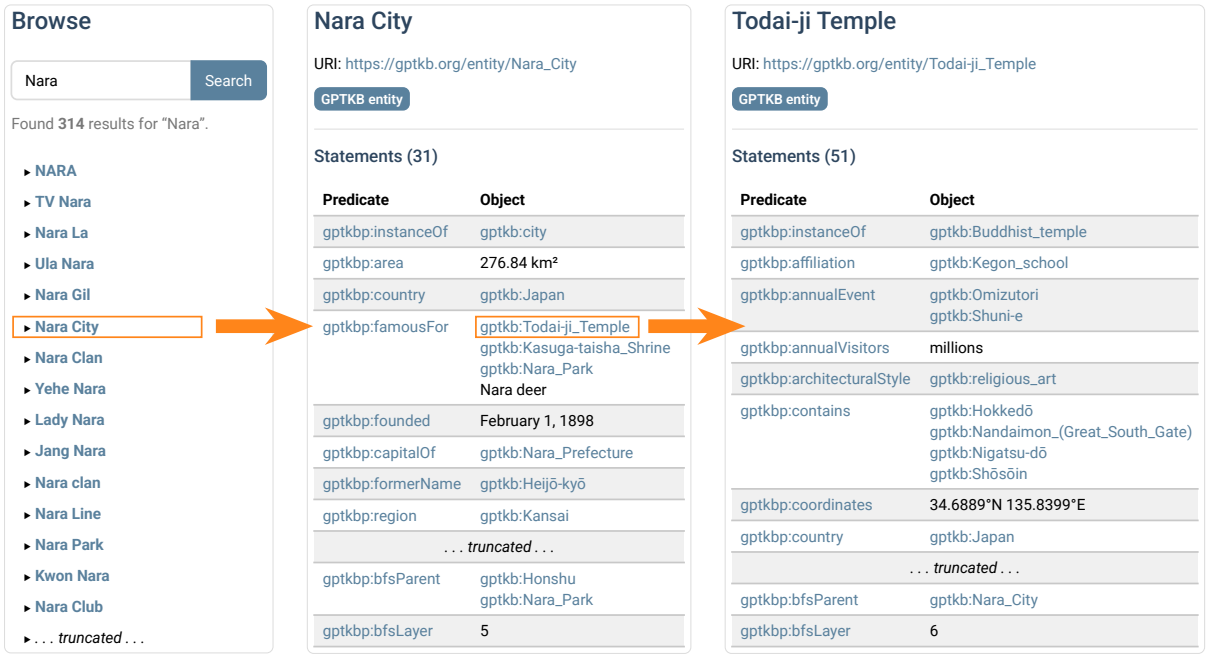

Figure 1: Searching an entity, and subsequent link-based exploration.

Firstly, GPTKB is hosted on the https://gptkb.org web server that provides a user interface to search entities via keyword queries and to perform link-based exploration to discover new connections and entities. Section 5.1 and Figure 1 provide a demonstration experience of this link-based exploration. The web interface is implemented by using the Python Django framework, and hosted on a Nginx web server. The KB is stored in an OpenLink Virtuoso server.

Secondly, we provide a SPARQL endpoint at https://gptkb.org/query/ that supports structured queries within a timeout window of 100 seconds.

Thirdly, we provide the RDF dump under the CC BY 4.0 license on the HuggingFace datasets library at https://huggingface.co/Knowledge-aware-AI.

5. Demonstration Experience

We divide the demonstration experience in two parts: (1) link-based KB exploration, and (2) structured SPARQL analytics and Wikidata comparison.

5.1. Link-based KB Exploration

Data about specific entities can be accessed via multiple routes:

1. The start page features a selection of direct links to entities such as *Vannevar Bush* and *San Francisco*.
2. The web portal features a search field in the top-right corner, which can be used for string-based search.
3. If an unambiguous entity name is known, one can directly access the entity's KB entry via the URL https://gptkb.org/entity/<NAME>.

Figure 1 shows how to initiate an entity search and continue with a link-based exploration. We start by typing this year's ISWC venue, *Nara*, which returns 314 results. Clicking on the *Nara City* result takes us to its entity page. The entity *Nara City* contains 31 statements, and a user can click on any of the entity objects, here for instance, *Todai-ji Temple*, to further explore connected entities.

Since each entity in GPTKB was identified as object from a parent entity during the BFS algorithm, we provide this information via the *bfsParent* relation. Additionally, the *bfsLayer* relation tells us at which layer knowledge elicitation was performed for the entity. In Figure 1, we learn that *Nara City* is a child entity of *Honshu*, and that its triples were elicited in layer 5. Clicking on the parent entity lets a user move up the layers of GPTKB.

5.2. SPARQL Querying for Analytics and Wikidata Comparison

A core feature of structured query languages is that they allow statistical analysis at scale. For this purpose, the GPTKB content is stored in a Virtuoso Triple store, whose content is exposed via a SPARQL query interface available at https://gptkb.org/query/. In the following, we show enabled analyses.

Most frequent classes Just what kind of entities does GPT know about? An overview is provided by the following query:

```
PREFIX gptkb: <https://gptkb.org/entity/>
PREFIX gptkbp: <https://gptkb.org/prop/>

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

The results are fundamentally different from, e.g., Wikidata, with a much stronger focus on digital artifacts (films, songs), and fiction.

Nationality bias Existing KBs as well as LLM training corpora are known to be Western- and English-language dominated [9], can this bias also be observed at the factual level of GPTKB? A quick glimpse can be obtained by counting the number of citizens per country known to GPTKB:

```
PREFIX gptkb: <https://gptkb.org/entity/>
PREFIX gptkbp: <https://gptkb.org/prop/>

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

o	ofreq
gptkb:American	374,263
British	133,940
gptkb:French	48,294
gptkb:German	45,381
gptkb:Indian	44,872
gptkb:Canadian	41,605
gptkb:Australian	30,978
Japanese	30,453

Notably, English language nationalities occupy the top places, at a much stronger bias than existing resources like Wikidata.

6. Related Work

Factual knowledge of LLMs is intensively researched, mostly via sample-based benchmarks or probes, such as the seminal LAMA probe by Petroni et al. [10]. However, these works typically draw sample from existing web resources, thereby introducing a confirmation bias that prevents the discovery of unexpected knowledge (or errors). For example, LAMA drew 50k triples from Wikidata, Wikipedia, and ConceptNet.

Few works have harvested LLM knowledge at scale. Nguyen and Razniewski [11] harvested one million commonsense assertions from BART and GPT-2, based on a pre-defined subject list. Cohen et al. [12] proposed to crawl factual LLM knowledge by recursively prompting them. Parović et al. [13] proposed domain-specific KB construction from LLMs, but did this only at the scale of a few hundred thousand entities. In this demo we build upon the GPTKB methodology by Hu et al. [1], which is a recursive methodology with judicious optimizations towards scalability, prompt-efficiency, and scoping, via parallelization, prompt-design, and dedicated NER.

Several large knowledge bases are deployed online [14], most notably Wikidata [2], Yago [3] and DBpedia [4]. Our web browsing and query interfaces are inspired by those.

In terms of LLM-generated datasets, the closest to ours might be Cosmopedia [15], an LLM-generated 25 billion token text corpus. However, Cosmopedia is intentionally designed to synthesize realistic-looking but invented texts, and has no goal of collecting factual LLM knowledge.

7. Conclusion

We have presented the https://gptkb.org web demonstrator, a knowledge base browser and query interface to GPTKB, a massive 100-million-triple KB built from GPT-4.1 using the GPTKB methodology [1]. Our demonstrator enables experimental insights into the potential of LLMs for complementing existing KB construction paradigms.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] Y. Hu, T.-P. Nguyen, S. Ghosh, S. Razniewski, Enabling LLM knowledge analysis via extensive materialization, in: ACL, 2025. URL: https://aclanthology.org/2025.acl-long.789.

[2] D. Vrandecic, M. Krötzsch, Wikidata: a free collaborative knowledge base, Commun. ACM 57 (2014). doi:10.1145/2629489.
[3] F. M. Suchanek, G. Kasneci, G. Weikum, Yago: a core of semantic knowledge, in: WWW, 2007. doi:10.1145/1242572.1242667.
[4] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. G. Ives, DBpedia: A nucleus for a web of open data, in: ISWC, 2007. doi:10.1007/978-3-540-76298-0_52.
[5] A. Fader, S. Soderland, O. Etzioni, Identifying relations for open information extraction, in: EMNLP, 2011. URL: https://aclanthology.org/D11-1142.
[6] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, et al., Never-ending learning, Communications of the ACM 61 (2018). doi:10.1145/3191513.
[7] J. Z. Pan, S. Razniewski, J.-C. Kalo, S. Singhania, J. Chen, S. Dietze, H. Jabeen, J. Omeliyanenko, W. Zhang, M. Lissandrini, et al., Large language models and knowledge graphs: Opportunities and challenges, Transactions on Graph Data and Knowledge (2023). doi:10.4230/TGDK.1.1.2.
[8] Y. Hu, T.-P. Nguyen, S. Ghosh, M. Müller, S. Razniewski, GPTKB v1.5: A massive knowledge base for exploring factual LLM knowledge, arXiv (2025). doi:10.48550/arXiv.2507.05740.
[9] G. Wenzek, M.-A. Lachaux, A. Conneau, V. Chaudhary, F. Guzmán, A. Joulin, É. Grave, CCNet: Extracting high quality monolingual datasets from web crawl data, in: LREC, 2020. doi:10.48550/arXiv.1911.00359.
[10] F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, A. Miller, Language models as knowledge bases?, in: EMNLP, 2019. doi:10.18653/v1/D19-1250.
[11] T.-P. Nguyen, S. Razniewski, Materialized knowledge bases from commonsense transformers, in: Workshop on Commonsense Representation and Reasoning (CSRR), 2022. doi:10.18653/v1/2022.csrr-1.5.
[12] R. Cohen, M. Geva, J. Berant, A. Globerson, Crawling the internal knowledge-base of language models, in: EACL Findings, 2023. doi:10.18653/v1/2023.findings-eacl.139.
[13] M. Parović, Z. Li, J. Du, Generating domain-specific knowledge graphs from large language models, in: Findings of the Association for Computational Linguistics: ACL 2025, 2025.
[14] G. Weikum, X. L. Dong, S. Razniewski, F. Suchanek, Machine knowledge: Creation and curation of comprehensive knowledge bases, Foundations and Trends in Databases 10 (2021). doi:10.1561/1900000064.
[15] L. B. Allal, A. Lozhkov, D. van Strien, Cosmopedia: A new frontier for open-source language models, 2024. URL: https://huggingface.co/blog/cosmopedia, Hugging Face blog.