MatKG-2: Unveiling precise material science ontology through autonomous committees of LLMs

Vineeth Venugopal
Massachusetts Institute of Technology
vineethv@mit.edu

Elsa Olivetti
Massachusetts Institute of Technology
elsao@mit.edu

Abstract

This paper introduces MatKG-2, a Material Science knowledge graph autonomously generated through a Large Language Model (LLM) driven pipeline. Building on the groundwork of MatKG, MatKG-2 employs a novel 'committee of large language models' approach to extract and classify knowledge triples with an established ontology. Unlike the previous version, which relied on statistical co-occurrence, MatKG-2 offers more nuanced, ontology-based relationships. Using open LLMs such as Llama2 7b and Bloom 1b/7b, the study offers reproducibility and broad community engagement. By using 4-bit and 8-bit quantized versions for fine-tuning and inference, MatKG-2 is also more computationally tractable and therefore compatible with most commercially available GPUs. Our work highlights the potential of MatKG-2 in supporting Material Science data infrastructure and in contributing to the semantic web.

1 Introduction

A knowledge graph (KG) is a structured representation of information that models the controlled vocabulary and ontological relations of a topical domain as nodes and edges. MatKG Venugopal et al. [2022], Venugopal and Olivetti [2023]is one of the first efforts to develop a comprehensive knowledge graph of material science autonomously through natural language processing (NLP). At the time of writing, MatKG expresses over 6.5 million relations between 200,000 entities comprising materials (MAT), their applications (APL), properties (PRO), characterization methods (CMT), synthesis methods (SMT), symmetry phase labels (SPL), and descriptors (DSC). This allows for querying of linked information, visualization of those links, as well as knowledge discovery through graph representation learning. As described previously, MatKG supports basic queries on materials such as "What are the applications and properties of TiO_2" or "What methods are used to characterize Cadmium Telluride" in a broad manner. Such semantically aware knowledge graphs aid the development of a robust data infrastructure in material science as well as the larger goal of a semantic web McCusker et al. [2020], Roch et al. [2018], Tshitoyan et al. [2019].

A relation in MatKG is expressed as the concatenation of the class tags of two entities; eg, the relation between (Fe_2O_3, Catalyst) is 'MAT-APL' and is qualified by the 'co occurrence frequency' - the number of documents in which the term pair occur simultaneously. In the absence of an established ontology in material science, these statistically motivated pseudo relations suggest correlations weighted by the co-occurrence frequency. As expected, these correlations often correspond to different ontological relations as illustrated by the triples (In_2O_3, MAT-APL, Optical material) and (Boron, MAT-APL, Nuclear reactor). While In_2O_3 'is used as' an Optical material, Boron 'is used in' a Nuclear reactor, demonstrating the inadequacy of a purely statistical co-occurrence mapping.

In this paper we demonstrate an autonomous large language model (LLM) knowledge graph generation pipeline that extracts knowledge triples (subject, relationship, object) from the abstracts of material science papers where the relationship is derived from an existing ontology and is hence

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

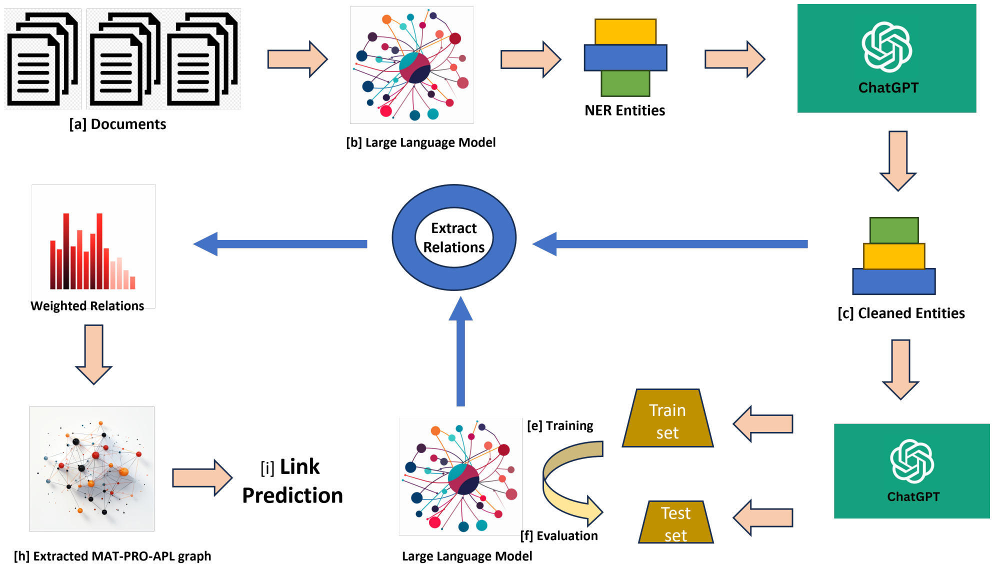


Figure 1: Knowledge graph pipline (a) Raw text corpus (b) named entity extraction through open LLMs (c) Cleaning and aggregation of entities using GPT 3.5 (d) Development of training/test set for relationship extraction using GPT 4.0 (e) Training of 4 bit and 8 bit quantized models for relationship extraction (f) Evaluation of these models (g) Extraction of relationships from the entity list (h) Construction of Graph (i) Link Prediction.

specific and more informative. Starting from text documents, fine-tuned open LLMs extract the controlled vocabulary based on the MatScholar schema listed above while a second set of LLMs classify the relationship between these entities into one among a previously defined set of relations. We introduce a novel 'committee of large language models' approach that simultaneously relies on the output of multiple LLMs to generate the final relation between two entities which is aggregated both by co-occurrence as well as by the performance of the LLMs on a test set. This leads to the construction of a new Knowledge Graph, MatKG-2, with carefully defined relationships.

We note that open LLMs such as Llama2 7bTouvron and et al [2023], Bloom 1b, and Bloom 7b Workshop et al. [2023] have been used for KG construction pipeline, which allows reproducibility and access to the community at large. In addition, the 4-bit and 8-bit quantized versions of these models have been used for fine tuning and inference, allowing for reproduction on most commercially available GPUs. Expensive LLMs like GPT 3.5 and 4.0 have only been used for smaller tasks such as training set development.

Finally, three common graph representation learning models - TransE, Distmult, and ComplEx Costabello et al. [2019] are used to predict new links in the graph thus constructed, thereby adding to the acquired knowledge base through transductive learning.

2 Methods

We downloaded 4.7 million abstracts of peer reviewed scientific papers from publisher sites through custom agreements and private APIs as described elsewhereKim et al. [2017]. These are stored in a local NoSQL database and form the raw text corpus for the KG construction pipeline. Fig 1 illustrates the components and processes involved in the process starting from (a) Raw text corpus (b) named entity recognition (NER) extraction through open LLMs (c) Cleaning and aggregation of entities using GPT 3.5 (d) Development of training/test set for relationship extraction using GPT 4.0 (e) Training of 4-bit and 8-bit quantized models for relationship extraction (f) Evaluation of these models (g) Extraction of relationships from the entity list (h) Construction of Graph (i) Link Prediction. The steps are described further below:

(b) Named Entity Recognition through open LLMs: The publicly available MatScholar NER dataset Weston et al. [2019]was used to fine tune Bloom 1b (4-bit), Bloom 7b (4-bit), Llama 2 (8-bit)

Table 1: Multilabel classification performance of models on test set

MAT - APL				
Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Bloom1b 4bit	78.68	79.10	78.68	78.44
Bloom7b 4bit	98.91	98.92	98.91	98.91
Llama2 8bit	71.38	73.79	71.38	70.66
MAT - PRO				
Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Bloom1b 4bit	65.26	66.46	65.25	64.61
Bloom7b 4bit	94.50	95.00	94.20	94.60
Llama2 8bit	56.46	56.59	54.30	54.30

using Quantized and Low Rank Adaptation (QLoRa) Dettmers et al. [2023]. The completions on the test set were evaluated using RogueL score and are given in the Table 3.

(c) Cleaning entities: The extracted entities are clubbed together based on a fuzzy Levenshtein metric which helps in grouping semantically similar entities *['electrode', 'electrodes', 'electroded']* and are subsequently replaced by a generalized entity *['Electrode']* selected by GPT 3.5. This results in a clean list of unique entities. This step is described in detail elsewhere Venugopal et al. [2022].

(d) Train/Test set development: After extracting relevant entities, we randomly choose 500 pairs of these entities. For each pair, we also select five Digital Object Identifiers (DOIs) where these entities are mentioned together in the abstract. We then use a predefined prompt (Appendix) to query GPT 4.0 via the OpenAI API. This query aims to identify the ontological relationship—denoted as R_{ab}—between each of the 500 pairs, using data from the five selected DOIs. GPT 4.0's most frequently identified relationship label for each entity pair (A, B) is taken as R_{ab}. If no label appears a majority of times (i.e., less than 3 out of 5), we assign 'None' as Rab for that pair. Finally, we split these 500 entity pairs into training and testing sets, maintaining an 80:20 ratio. This results in a training set of 400 pairs and a testing set of 100 pairs.

(e -f) Training and Evaluation: The train set from (d) is used to train Bloom 1b (4-bit), Bloom 7b (4-bit) and llama 2 (8-bit) using Q-Lora. The parameters for training are given in the appendix. The performance of the models on the test set are given in Table 1. All models achieve a weighted f1 score > 0.5 across the two entity pairs chosen in the study.

(g) Model Inference: The trained models are subsequently deployed in inference mode to discern relationships among all entity pairs in MatKG. We focus here on Material-Application (MAT-APL) and Material-Property (MAT-PRO) pairs. For each of these categories, we randomly select 10 DOIs corresponding to 1,000 entity pairs. The outcome is a Material-Application-Property (MAT-APL-PRO) graph. Importantly, the methodology is adaptable and can be used for other types of entity pairs, spanning multiple ontological categories.

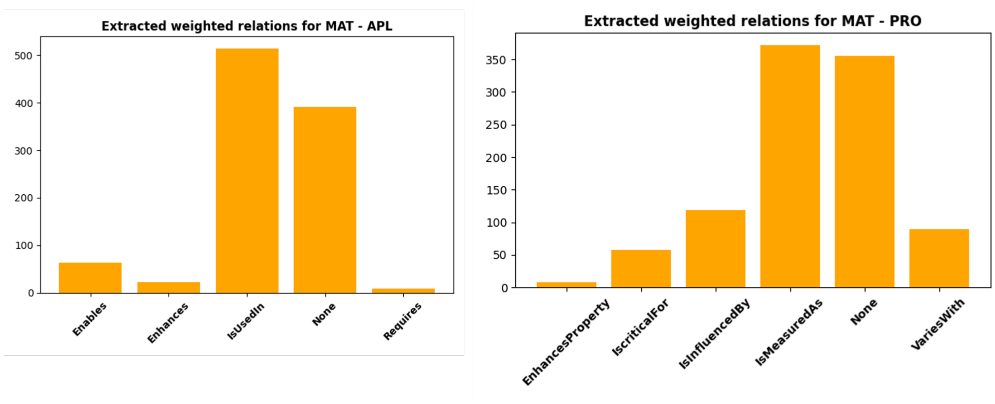
For a given entity pair (A, B), we amass a total of 30 extracted relations—10 from each sampled DOI abstract, multiplied by the three different models used. For each specific instance (A, B, DOI), the output from each model is weighted by its F1 score as per the test set results shown in Table 1. This weighting approach prioritizes the relationships identified by the models that demonstrated superior performance in the test phase. These weighted results are aggregated across all 10 DOI samples to yield the final ontological relationship, denoted as Rab, between entities (A, B). The number of final relationships in the aggregated graph are shown in the bar plot on Fig 2. This multi-model, multi-DOI approach could mitigate the 'hallucination' issue often encountered with LLMs, providing opportunity for a more reliable and robust ontological mapping.

(h-i) Graph construction and Link prediction: We construct a graph G using NetworkX in Python, based on extracted knowledge triples [subject,relationship,object]. To predict new links, we employ TransE, DistMult, and ComplEx models. TransE translates the head to the tail entity through the relation vector. DistMult employs bilinear product to model triples. ComplEx uses complex-valued embeddings to capture both symmetric and asymmetric relationships. These models are trained using existing triples in G, aiming to predict the likelihood of new relationships.

Table 2: Performance of KG Embedding models on test set

Model	**MRR**	**Hits@10**
TransE	0.77	0.60
ComplEx	0.70	0.47
Distmult	0.61	0.47

Figure 2: Extracted weighted relations for (a) Material - Applications (b) Material - Property



3 Results

Fig 2 shows the extracted results for the sample of 1000 MAT-APL and MAT-PRO entity pairs The most common relationship extracted by the models between materials and application is 'IsUsedIn' while that for materials and properties are 'IsMeasuredAs'. A significant number of entity - roughly 30 percent in each set - is classified as 'None', implying that the models cannot agree/identify on a single relationship between them.

We hypothesize that the multi-model, multi-DOI approach mitigates the hallucination issue by introducing a form of ensemble learning and diversified sourcing. By using different LLMs and weighting their results based on their performance in a test set, the methodology reduces the likelihood that a spurious or incorrect relationship is accepted. Furthermore, sampling from multiple DOIs adds another layer of verification, making it less probable that a hallucinated or erroneous relationship from a single model or source will dominate the final results. While this does represent a significant step forward in the autonomous generation of knowledge graphs, these relationships need to verified and a randomized manual sampling is part of our future work. In addition, the quality of the training data can also be improved through a human in the loop based approach.

To further quantify the impact of the ensemble models on hallucination, we randomly samples 100 predicted labels from the inference data and manually annotated them. The results are shown in table 3 and show that the ensemble model does perform better than any of the models taken individually. This supports our hypothesis that a committee of LLMs fine tuned on the same task performs better than any single LLM.

The generated graph, G has 2670 nodes and 1253 edges. As a proof of concept, G represents a first of it's kind knowledge graph, one that is autonomously created but is also semantically rich and ontologically well defined.

The Ampligraph library Costabello et al. [2019] was used to train three graph representation models (TransE, Distmult, CompleX) with 150 dimensional embeddings to model the vector space of entities, through which link prediction is performed on a test set. The TransE Bordes et al. [2013] model was found to perform the best based on the Mean Reciprocal Rank on the test set as seen in Table 2.

Table 3: Agreement of individual models and the model ensemble on a random collection of 100 entity pairs

	Bloom 1b 4 bit	Bloom 7b 4bit	Llama2 8bit	Ensemble
MAT-APL	72	79	61	81
MAT-PRO	73	81	50	81

4 Broader Impact

MatKG-2 is the first step towards the complete synthesis of materials knowledge that allows for the richer databases not just for materials but also for applications, properties, and characterization methods. By using a committee of LLMs we hypothesize that the issue of hallucination can be addressed, as well as by clubbing across different documents.

5 Data and Code

The MatKG-1 dataset is available in both a CSV and RDF format in `https://doi.org/10.5281/zenodo.10022726` Venugopal and Olivetti [2023].

The github repository Venugopal [2023] contains detailed tutorial style notebooks that demonstrate the usage of MatKG-1.

References

Vineeth Venugopal, Sumit Pai, and Elsa Olivetti. Matkg: The largest knowledge graph in materials science – entities, relations, and link prediction through graph representation learning, 2022.

Vineeth Venugopal and Elsa Olivetti. Matkg 1.2, October 2023.

James P McCusker, Neha Keshan, Sabbir Rashid, Michael Deagen, Cate Brinson, and Deborah L McGuinness. Nanomine: A knowledge graph for nanocomposite materials science. In *International Semantic Web Conference*, pages 144–159. Springer, 2020.

Loïc M Roch, Florian Häse, Christoph Kreisbeck, Teresa Tamayo-Mendoza, Lars PE Yunker, Jason E Hein, and Alán Aspuru-Guzik. Chemos: orchestrating autonomous experimentation. *Science Robotics*, 3(19):eaat5559, 2018.

Vahe Tshitoyan, John Dagdelen, Leigh Weston, Alexander Dunn, Ziqin Rong, Olga Kononova, Kristin A Persson, Gerbrand Ceder, and Anubhav Jain. Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature*, 571(7763):95–98, 2019.

Hugo Touvron and et al. Llama 2: Open foundation and fine-tuned chat models, 2023.

BigScience Workshop, :, Teven Le Scao., and et al. Bloom: A 176b-parameter open-access multilingual language model, 2023.

Luca Costabello, Sumit Pai, Chan Le Van, Rory McGrath, Nicholas McCarthy, and Pedro Tabacof. AmpliGraph: a Library for Representation Learning on Knowledge Graphs, 2019. URL `https://doi.org/10.5281/zenodo.2595043`.

Edward Kim, Kevin Huang, Stefanie Jegelka, and Elsa Olivetti. Virtual screening of inorganic materials synthesis parameters with deep learning. *npj Computational Materials*, 3(1):1–9, 2017.

Leigh Weston, Vahe Tshitoyan, John Dagdelen, Olga Kononova, Amalie Trewartha, Kristin A Persson, Gerbrand Ceder, and Anubhav Jain. Named entity recognition and normalization applied to large-scale information extraction from the materials science literature. *Journal of chemical information and modeling*, 59(9):3692–3702, 2019.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

Table 4: ROUGE-L scores for LLMs on NER task compared with ground truth

Model	ROUGE-L
Bloom1b 4bit	0.62
Bloom7b 4bit	0.61
Llama2 8bit	0.55

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795, 2013.

Vineeth Venugopal. Matkg, 2023. URL `https://github.com/olivettigroup/MatKG`.

6 Appendix

6.1 Prompt Template for MAT - PRO

Please describe the relationship between carbon and mobility based on the following text,using only one of the following relationships: 'EnhancesProperty','IsMeasuredAs', 'VariesWith', 'IsCriticalFor', 'IsInfluencedBy', or 'None'

Text: Organic proton-conducting molecules are presented as alternative materials to state-of-the-art polymers used as electrolytes in proton-exchanging membrane (PEM) fuel cells. Instead of influencing proton conductivity via the mobility offered by polymeric materials, the goal is to create organic molecules that control the proton-transport mechanism through supramolecular order. Therefore, a series of phosphonic acid-containing molecules possessing a carbon-rich hydrophobic core and a hydrophilic periphery was synthesized and characterized. Proton conductivity measurements as well as water uptake and crystallinity studies (powder and single-crystal X-ray analysis) were performed under various conditions. These experiments reveal that proton mobility is closely connected to crystallinity and strongly dependent on the supramolecular ordering of the compound. This study provides insights into the proton-conducting properties of this novel class of materials and the mechanisms responsible for proton transport.

Output:

6.2 Prompt Template for MAT - APL

Please describe the relationship between Nafion and proton conductivity based on the following text, using only one of the following relationships: 'Enables', 'Enhances', 'IsUsedIn', 'Requires' or 'None'

Text: Here we demonstrate enhanced proton conduction through polyelectrolyte matrices comprising nano- peapods of phosphotungstic acid (PWA) filled carbon nanotubes (CNTs). The ionic nanopeapods were found to provide rapid proton conduction pathways to design nanocomposite proton exchange membranes (PEMs) for high-performance fuel cell applications. Nanopeapod (0.5 wt) incorporated Nafion- based PEMs offer improved proton conductivity, especially at elevated temperatures and low-humidity (0.202 S cm 1 compared with 0.132 S cm 1 for recast Nafion membrane at 90 °C), and about four times higher maximum power density at 40 pc R.H. and 120°C (302 mWcm 2 vs. 84 mW cm 2 for recast Nafion).

Output:

6.3 Paramaters for fine tuning relationship classification models

quantization:
bits: 4

```
generation:
temperature: 0.1
max new tokens: 512

epochs: 25
batch size: 1
eval batch size: 1
enable gradient checkpointing: true
gradient accumulation steps: 16
learning rate: 0.00001
optimizer:
type: paged adam
params:
eps: 1.e-8
betas:
- 0.9
- 0.999
weight decay: 0
learning rate scheduler:
warmup fraction: 0.03
reduce on plateau: 0
```