We can reliably update **Knowledge Graphs** from **News Articles** using **LLMs** and **GraphRAG**

GitHub Repository

Building and Maintaining Knowledge Graphs of Company-Information using Large Language Models and Graph Retrieval Augmented Generation

Jakob Schmidbauer, Mikhail Azaryan
Project Supervisor: Tim Schwabe

Motivation

- Many tasks, such as Graph Neural Networks (GNNs) and Graph Retrieval Augmented Generation (GraphRAG) rely on **accurate and up-to-date Knowledge Graphs** (KGs).
- As the world is constantly changing, existing KGs need to be constantly updated to remain useful.
- For this project, we have developed algorithms that first **build a KG from Wikidata.org** and then **update it based on new information from scraped news articles**.
- Our KG of company data supports **time slicing** and can be used to train a GNN to attempt stock price predictions.
- The underlying updating algorithms are not specific to company data and can be **generalised to other domains**.

Building an Initial KG from Wikidata.org

- Starting with a list of company names, the algorithm crawls Wikidata.org entries using breadth-first search, iteratively adding nodes and relationships.
- The algorithm supports setting start and end dates, variable search depth, node types to include, and a maximum branching factor.
- We use neo4j as our graph database, as it handles large relational datasets and supports complex queries.

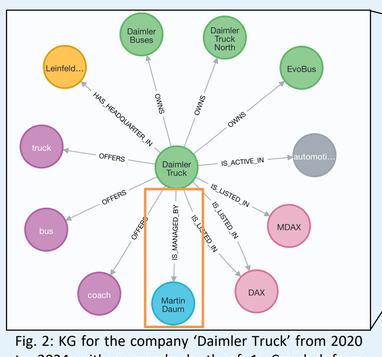


Fig. 2: KG for the company 'Daimler Truck' from 2020 to 2024 with a search depth of 1. Crawled from Wikidata.org as of Jan 20, 2025.

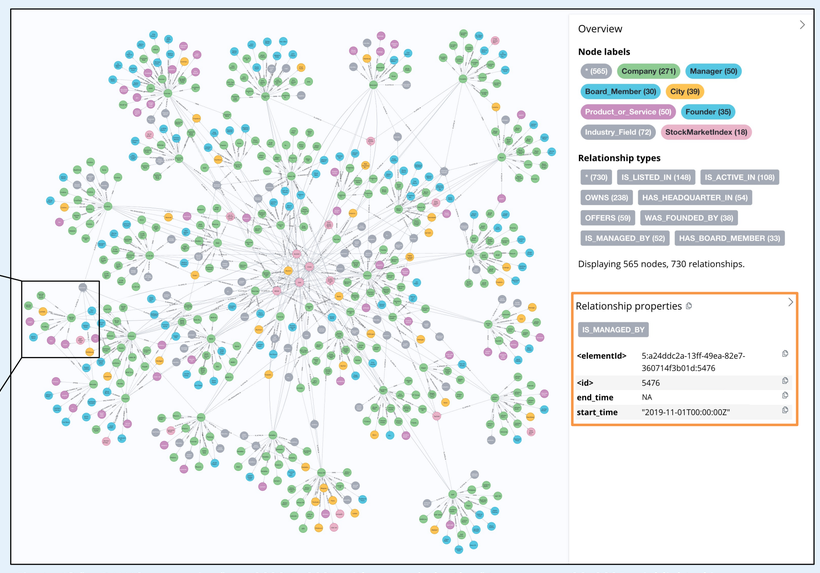
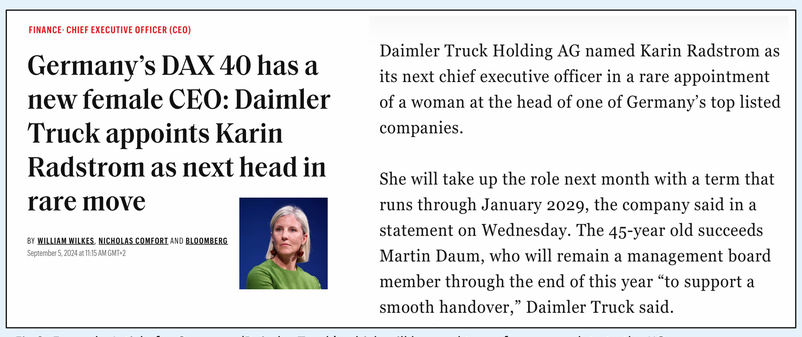


Fig. 1: KG for DAX40 companies from 2020 to 2024 with a search depth of 1 on January 20, 2025. Highlighted in orange are the relationship properties of the former Manager *Martin Daum*, as this will serve as an example for the updating process.

Scraping News Articles

- Next, we scrape articles using the New York Times Article Search API, including metadata such as title, text, publication date, keywords and related articles.
- We then filter for articles that describe a change in the relationship between entities in the KG.
- Finally, we use an LLM to summarise the articles into short sentences with the most important information.



FINANCE- CHIEF EXECUTIVE OFFICER (CEO)

Germany's DAX 40 has a new female CEO: Daimler Truck appoints Karin Radstrom as next head in rare move

BY WILLIAM WILKES, NICHOLAS COMFORT AND BLOOMBERG
September 5, 2024 at 11:15 AM GMT+2

Daimler Truck Holding AG named Karin Radstrom as its next chief executive officer in a rare appointment of a woman at the head of one of Germany's top listed companies.

She will take up the role next month with a term that runs through January 2029, the company said in a statement on Wednesday. The 45-year old succeeds Martin Daum, who will remain a management board member through the end of this year "to support a smooth handover," Daimler Truck said.

Fig 3: Example Article for Company 'Daimler Truck', which will be used to perform an update to the KG.
Source: fortune.com/europe/2024/09/05/germany-has-a-new-female-ceo-daimler-truck-appoints-karin-radstrom-as-the-next-head-dax/

Updating the KG

- Algorithmic steps:
  - Determine which parts of the KG are relevant to the article.
  - Retrieve the relevant information from the KG as a list of triples.
  - Provide the relevant triples and the pre-processed article to the LLM to obtain graph update proposals as new lists of *unchanged*, *added* and *deleted* triples.
  - Convert the triples into neo4j/cipher update queries.
  - Execute the queries and validate the result.

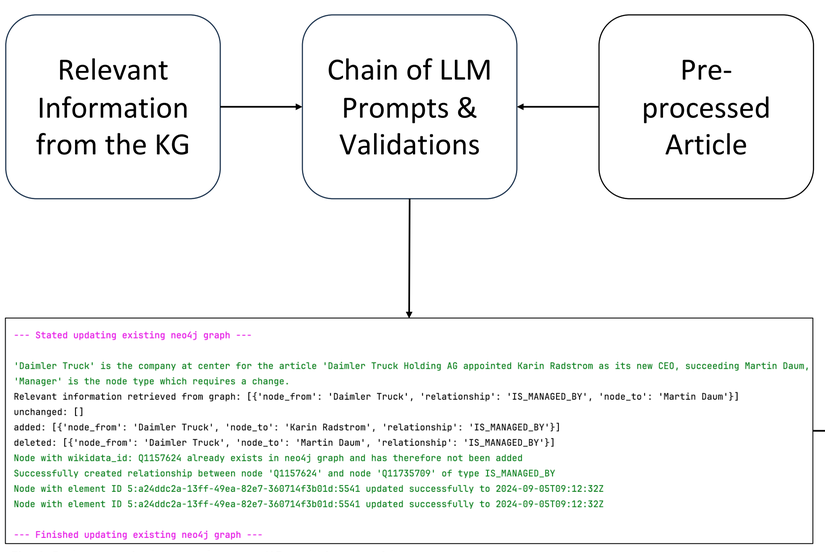


Fig. 4: Python terminal output from our KG-updating-algorithm

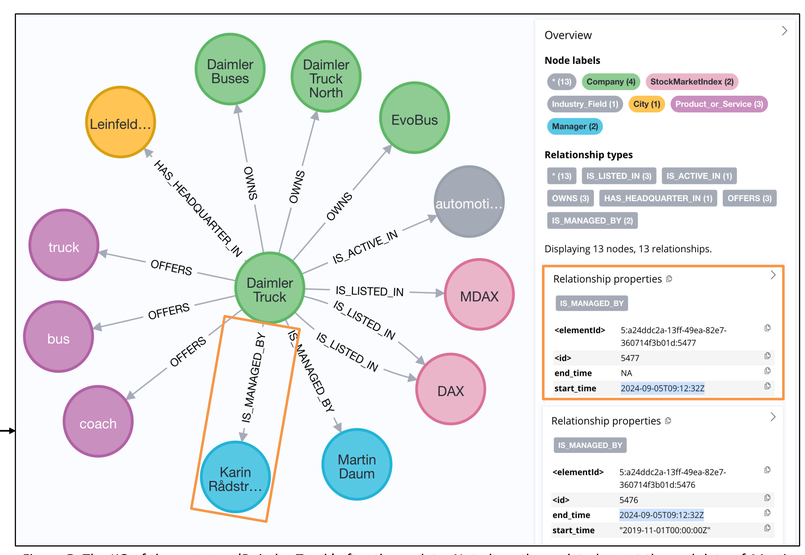
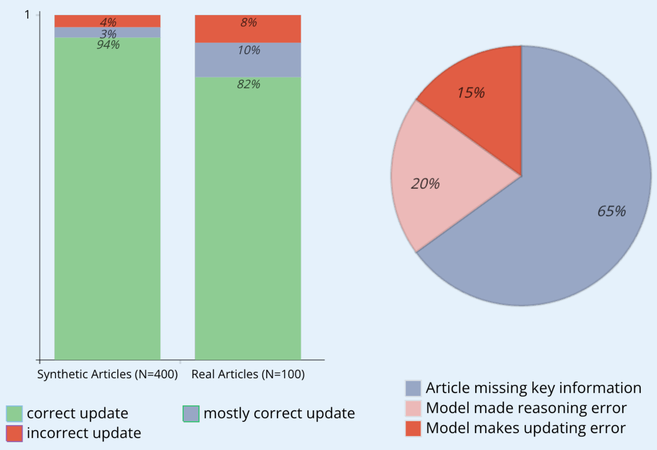


Figure 5: The KG of the company 'Daimler Truck' after the update. Note how the update has set the end date of *Martin Daum's* IS_MANAGED_BY relationship to Sep 2024 and added *Karin Radstrom* as the new manager from that date.

Benchmarking



Key Takeaways

- **Updating the KG works well**, especially when the articles contain all relevant information.
- Decomposing the problem into **separate reasoning steps with output constraints** makes the LLM models more reliable and faster but limits updates to a predefined set of changes.
- **Updating scales well (almost linearly)** with total graph size, but worse with high branching factors.
- We have built a **dataset** of DAX30 companies with **more than 12000 nodes**. The algorithm works for every company and stock index on Wikidata.org

Next Steps

- Further improve the accuracy of updates by combining multiple news sources and online searches, implementing best-of-n sampling for the LLM and improved validation of update results.
- Retrain/Fine-tune a LLM to work better with KG input data and articles to update the KG.
- Train a GNN on the dataset to retrieve stock price predictions from news articles.

Technical University of Munich, TUM School of Computation, Information and Technology, Bachelor Practical Course: Data Engineering

Jakob.Schmidbauer@tum.de
Mikhail.Azaryan@tum.de