GenKD: Generative Knowledge Discovery through Knowledge Graphs and Large Language Models

Fouad Zablith[1,*], Shadi Youssef[1] and Mathieu d'Aquin[2]

[1]*Suliman S. Olayan School of Business, American University of Beirut, PO Box 11-0236, Riad El Solh, 1107 2020, Beirut, Lebanon*
[2]*LORIA, Université de Lorraine, CNRS, Nancy, France*

Abstract

With the continuous growth of data published on the web, knowledge discovery is getting increasingly challenging. This challenge is mainly driven by the knowledge discovery process that often requires the continuous aggregation and exploration of questions and patterns that span local and external knowledge sources. This work investigates the facilitation of knowledge discovery over distributed sources of knowledge on the web. We present GenKD, a Generative Knowledge Discovery framework that leverages the semantic interconnectedness of knowledge graphs and the generative capabilities of Large Language Models (LLMs). GenKD enables, through a human-AI collaborative process, the automatic generation of relevant questions, executable queries, and visualizations to uncover patterns from local and external knowledge graph sources. We demonstrate the feasibility of the proposed framework through a case study in the context of bee colonies and stressors.

Keywords

Knowledge discovery, knowledge graphs, large language models, data exploration, artificial intelligence, human-AI collaboration

1. Introduction

Knowledge discovery and data exploration often involve manipulating existing data to discover interesting insights [1, 2]. Insights usually emerge through an iterative process of pattern detection and question answering over the available data, which are often distributed among local and external sources [3]. Such sources have been increasingly available on the web in the form of knowledge graphs and linked data [4]. As a result, research on knowledge exploration on the web of data has been advancing in several directions [5]. For example, some related research focused on query exploration by example [6]. Other works studied how to enrich and augment knowledge graphs through manual and automatic entities linking and alignment [4, 7, 8, 9]. More recently, there has been an increased interest in leveraging Large Language Models (LLMs) to support data visualization [10, 11, 12] and SPARQL query generation from natural language [13, 14]. However, it is still challenging for data analysts to manage local and external knowledge sources to generate relevant questions, construct answerable queries from the combined data, and design visualizations that support their discovery endeavors.

This demo paper investigates the following research question: How can we facilitate the knowledge discovery process over distributed sources of knowledge on the web? We present GenKD, a novel Generative Knowledge Discovery framework that leverages the semantic data connections of knowledge graphs and the generative capabilities of Large Language Models (LLMs). GenKD supports, through a human-AI collaborative process, the automatic generation of relevant questions, executable queries, and data visualizations over local and external knowledge graph sources. A demo is implemented in the context of bee colonies and stressors in the US to illustrate the feasibility of the proposed framework. It includes a web-accessible prototype designed to enable a user guided LLM process to generate questions, queries, and visualizations that are adaptive to the local and external knowledge graph contexts. This work contributes to advancing knowledge discovery at a web scale through a human guided

ISWC 2025 Companion Volume, November 2–6, 2025, Nara, Japan

[*]Corresponding author.

🖂 fouad.zablith@aub.edu.lb (F. Zablith); say09@mail.aub.edu (S. Youssef); mathieu.daquin@loria.fr (M. d'Aquin)
🌐 https://fouad.zablith.org/ (F. Zablith)
iD 0000-0002-8978-9911 (F. Zablith); 0000-0001-7276-4702 (M. d'Aquin)

process building on the semantically rich capabilities of knowledge graphs and the generative features of LLMs.

2. Background

Knowledge discovery and exploration is an iterative process that involves several steps [1, 3]. It usually starts with one or several data sources, depending on the data analyst's goals and context, followed by processing and manipulating the data to identify interesting patterns and insights [1]. Knowledge graphs have been providing substantial opportunities for organizing data at a web scale [4, 15]. They are seen as enablers of exploratory data analytics tasks in general, and more specifically at the level of exploratory search tasks [16]. Existing efforts have focused on supporting analysts in exploring their knowledge graph data by example [6]. They offer the potential to reverse engineer knowledge graph SPARQL queries based on user-provided examples. However, with the interconnected nature of knowledge graphs, it becomes more challenging for analysts to anticipate the potential questions that may be answered by their local data sources, especially when combined with other sources.

Uncovering interesting questions during the knowledge discovery process often requires the fusion of multiple knowledge sources. To answer such questions, knowledge graphs that focus on certain contexts must be augmented with additional information. Aligning and mapping knowledge entities is a core task to enrich and construct knowledge graphs [4]. This entity linking aspect can be done using tools for linking graphs [7, 8] and by relying on the entities' external identifiers and manual efforts [9]. We observe that such efforts focus on the backend aspects of linking and mapping knowledge graphs. However, knowledge discovery requires human guidance and several iterations. This process expects analysts to ask the right questions based on the data, and to develop queries that can be executed to fetch and visualize the data to discover insights [3].

Recently, we have witnessed an increased adoption of Large Language Models such as ChatGPT, Claude, and Gemini to perform a wide range of tasks [17]. For example, LLMs have been investigated in the automatic generation of data visualizations [10, 11, 12]. In the context of knowledge graphs, there are increased attempts to leverage language models in accessing knowledge. For example, some efforts were invested in supporting the generation of queries from natural language [13, 14]. In this context, it is assumed that analysts are aware of the questions that they are aiming to answer, or of the potential questions that they may be able to answer through the knowledge graphs. This may not be often the case, in situations where the analyst would need more guidance in the question generation process, which may lead to serendipitous findings [18]. Furthermore, another challenge is at the level of combining the existing datasets with external knowledge sources to increase the potential richness of findings aligned with the analysts' needs. We see an opportunity to further study LLMs and knowledge graphs' capabilities in advancing knowledge discovery on the web of data.

3. Proposed Approach: GenKD Framework

Building on the knowledge discovery process [1], we propose GenKD, a Generative Knowledge Discovery framework that aims to support users to better collaborate with AI during the knowledge discovery process. Figure 1 illustrates the framework's components. One of the challenges in knowledge discovery is to understand the data available and the potential of extending such data with relevant external sources that may fill some knowledge gaps. Understanding the context of analysis is not only relevant to the analyst, but is also a key requirement for guiding and prompting the LLM with appropriate knowledge entities. The *Context Generator* component aims at representing the local context from the user's knowledge graph, with the potential extension to external knowledge context sources such as the linked open data cloud. Knowledge graphs facilitate the generation of such contexts on the fly through a series of SPARQL queries sent to local and external endpoints.

The generated context in the form of knowledge graph entities is combined into prompts using the *Prompt Builder* component. The prompts in the framework include question prompts, query prompts,

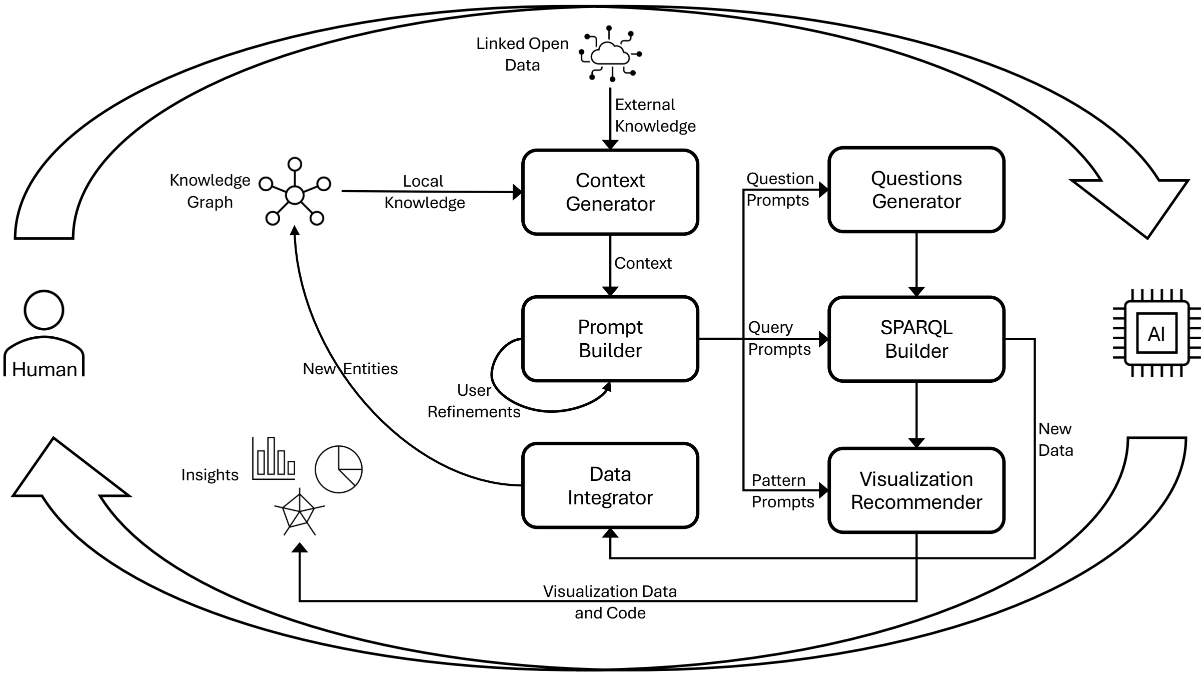


Figure 1: Generative Knowledge Discovery Framework.

and pattern prompts. Question prompts are designed to guide the LLM to generate questions that are relevant to the local and external knowledge contexts through the *Questions Generator* component. A clearly articulated context (local and external) plays an important role in crafting prompts that assist the LLM to generate questions that are answerable by the data. The suggested questions are then used to generate the SPARQL queries through the *SPARQL Builder* component. At this level, query prompts are designed to construct SPARQL queries that can be executed against both local and external knowledge graph endpoints. The resulting data is then transformed into a format (e.g., tabular) that can be easily accessible and processed by the *Visualization Recommender* component. At this level, pattern prompts are used for the LLM to select data that may contribute to interesting patterns (e.g., highly correlated data fields) and suggest appropriate visualizations according to the data types and question scopes. In addition to the choice of visualizations, the LLM is prompted to generate the visualization code and required libraries to use in the visualization process. While the framework provides predesigned prompts, it supports human-AI collaboration by allowing users to guide the LLM through custom prompts according to their needs with respect to the selection of knowledge graph entities, questions generated, queries suggested, and visualizations recommended.

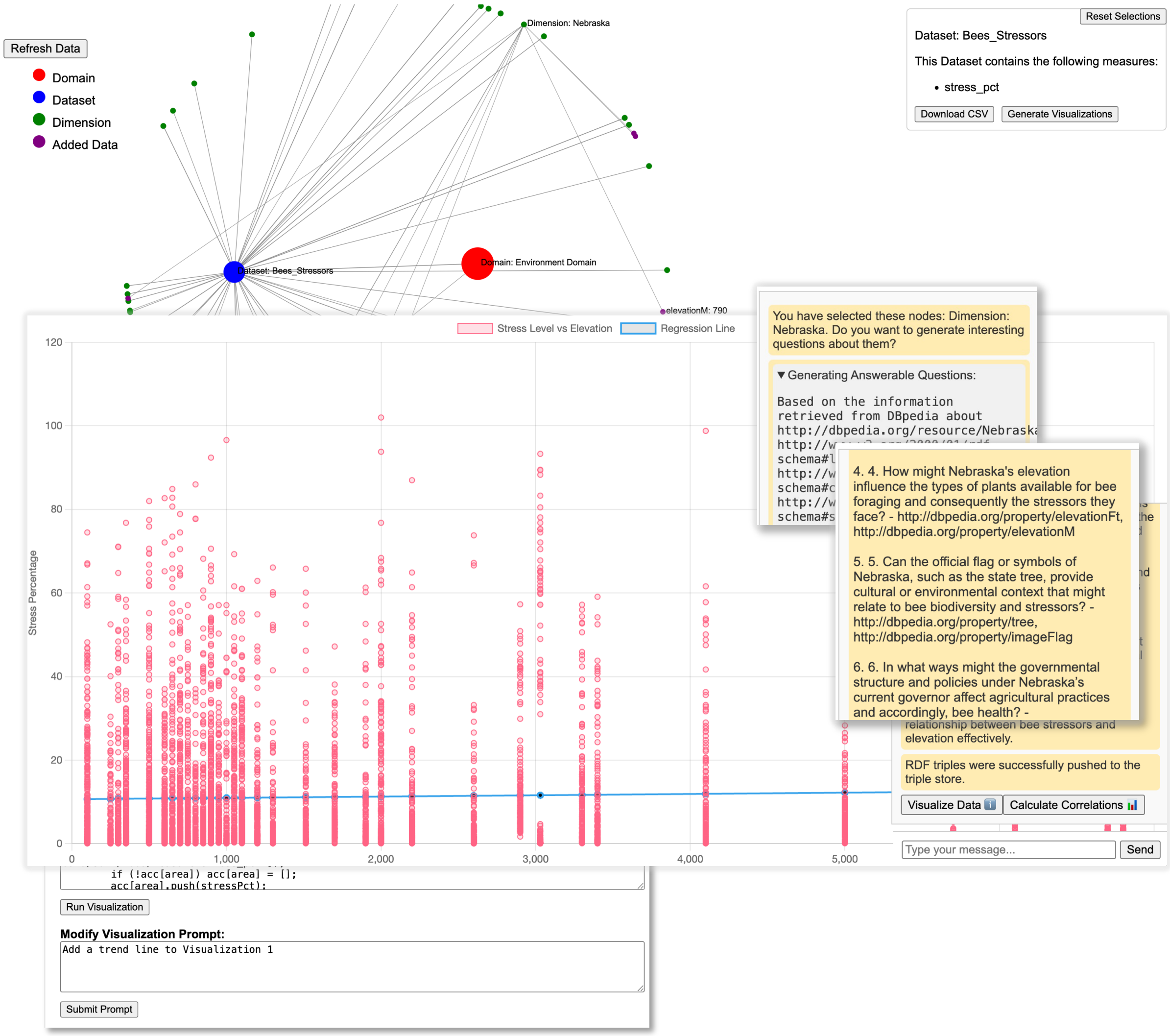
4. GenKD Demo: a Use Case on Bee Colonies in the US

We demonstrate the feasibility of the proposed framework through a case study in the context of exploring a knowledge graph in the domain of analyzing statistics about bee colonies. The data contains information about bee colonies (i.e., number of colonies, colonies lost or added, their percent lost, etc.) and bee stressors (i.e., stress type, month range, percent impacted by stress, etc.) in the United States[1]. We built a prototype of the GenKD framework based on a JavaScript application that can be loaded directly in a browser[2]. An OpenAI key is needed to enable the LLM features. A video recording that showcases the key demo features is also available[3]. Figure 2 shows details of the main application

[1] The data was transformed into a knowledge graph from the following data source: https://github.com/rfordatascience/tidytuesday/blob/main/data/2022/2022-01-11/readme.md

[2] The GenKD prototype is available online: https://linked.aub.edu.lb/apps/genkd

[3] The video recording is available at: https://youtu.be/Cnj05zOz1pY



Dataset: Bees_Stressors

This Dataset contains the following measures:

- stress_pct

You have selected these nodes: Dimension: Nebraska. Do you want to generate interesting questions about them?

▼ Generating Answerable Questions:

```
Based on the information
retrieved from DBpedia about
http://dbpedia.org/resource/Nebraska
http://w
schema#l
http://w
schema#c
http://w
schema#s
```

4. 4. How might Nebraska's elevation influence the types of plants available for bee foraging and consequently the stressors they face? - http://dbpedia.org/property/elevationFt, http://dbpedia.org/property/elevationM

5. 5. Can the official flag or symbols of Nebraska, such as the state tree, provide cultural or environmental context that might relate to bee biodiversity and stressors? - http://dbpedia.org/property/tree, http://dbpedia.org/property/imageFlag

6. 6. In what ways might the governmental structure and policies under Nebraska's current governor affect agricultural practices and accordingly, bee health? -

relationship between bee stressors and elevation effectively.

RDF triples were successfully pushed to the triple store.

```
        if (!acc[area]) acc[area] = [];
        acc[area].push(stressPct);
```

Modify Visualization Prompt:

```
Add a trend line to Visualization 1
```

Figure 2: Screenshots of the GenKD Demo Features Applied on the Bees Case.

features.

Users can choose an existing knowledge graph to explore, or specify an endpoint where their local knowledge graph is hosted[4]. The application then loads the data and visualizes it as a graph for the user to interact with. The user can start the discovery process by clicking and expanding the nodes of interest. In the background, the context generator component captures the context of the knowledge graph by recording the nodes clicked by the user. For example, in Figure 2, the user clicked a node path that included the *Bees Stressor* dataset and the *Nebraska* dimension instance. After the user finishes interacting with the data, the prompt builder component combines the clicked nodes with potential anchors from external linked open data endpoints. The DBpedia endpoint is used in this demo. The external context is fleshed out by traversing the related entities on DBpedia. For example, the state Nebraska is anchored in the `https://dbpedia.org/page/Nebraska` resource, and related entities (e.g., the `dbo:areaLand` or `dbp:areaTotalKm`) and combined to generate the external context. The *Prompt Builder* component takes the combined contexts and generates question prompts that are passed to the LLM using the OpenAI API. Having explicit semantic contexts with full DBpedia and local URIs passed to the LLM increases the chances of having queries answerable by the endpoints.

The LLM generates a set of questions that are relevant to the context, as shown in Figure 2. For example, one of the proposed questions by the AI model was: *How might Nebraska's elevation influence*

[4] In the current demo implementation, the endpoint is required to follow the RDF Data Cube vocabulary (https://www.w3.org/TR/vocab-data-cube/) to control access to the data measures and dimensions

the types of plants available for bee foraging and consequently the stressors they face? It is worth noting that the proposed question by the LLM can't be answered directly by the local knowledge graph on bees, as it doesn't contain the *elevation* of Nebraska. This exploratory new question, which is a core feature of the framework, was enabled by the external context derived from the online DBpedia endpoint.

The user can select the proposed question by the AI model that may be of interest, and then the *SPARQL Builder* component is triggered through a *Query Prompt*. The prompts at this level are designed to generate executable queries that can fetch the needed data to answer the question. For example, to answer this specific question, the elevation of Nebraska needs to be extracted from the DBpedia endpoint. The LLM then automatically generates the SPARQL query, with the option to push the new data to the local knowledge graph through the *Data Integrator* component. In the following step, the user can trigger the *Visualization Recommender* component that automatically checks potential correlations between data entities, generates the visualization code through the LLM, and renders the visualization in the application using JavaScript. The lower part of Figure 2 shows the generated visualization of a scatter plot proposed by the LLM to visualize the relationship between the elevation of states and the bees' stress percentage. The user can manipulate the visualization by further prompting the LLM, without the need to write any code. Table 1 shows a sample of prompts created by the prompt builder component.

Question Prompt Based on the information retrieved from DBpedia about *http://dbpedia.org/resource/Nebraska: http://www.w3.org/2000/01/rdf-schema#label, http://www.w3.org/2000/01/rdf-schema#comment, http://www.w3.org/2000/01/rdf-schema#seeAlso <...remaining related entities to the resource...>* Suggest relevant questions that can be answered from dbpedia about: *http://dbpedia.org/resource/Nebraska*. Make sure to include the DBpedia property that answers each question directly after each question (without line break). Make the questions relevant to the dataset: *http://linked.aub.edu.lb/data/Bees/Dataset/Bees_Colonies* and to the domain: *http://linked.aub.edu.lb/data/Bees/Domain/Environment_Domain* and are able to provide added value, avoid too general questions that might not be related to the dataset or to the topic. The node is connected to: *http://linked.aub.edu.lb/data/Bees/observation/Bees_Stressors-Disesases-Nebraska-April-June-2021...* Please ensure that the questions are answerable using DBpedia's data. Avoid questions that require information beyond what DBpedia provides or questions that are too general. If no specific questions are possible, provide related general questions that are still within the scope of DBpedia's knowledge base.
Query Prompt Generate a SPARQL query that can be executed on DBpedia to answer this question: "What is the elevation range in Nebraska, and how might this affect the stress factors on bee populations in the state? http://dbpedia.org/property/elevationFt, http://dbpedia.org/property/elevationM" note that the relevant DBpedia property(s) is with the question, avoid querying other things, if it cannot answer the question, let me know.
Pattern Prompt You are an expert JavaScript developer and data visualization designer. Given these CSV headers: *Observation URI, http://purl.org/linked-data/cube#dataset, http://purl.org/linked-data/sdmx/2009/dimension#refArea, http://purl.org/linked-data/sdmx/2009/dimension#refPeriod, http://purl.org/linked-data/sdmx/2009/measure#stress_pct, http://purl.org/linked-data/sdmx/2009/measure#colony_added, http://purl.org/linked-data/sdmx/2009/measure#colony_lost ...* generate a complete JavaScript function using Chart.js that takes parsed CSV data as input and renders at least 6 diverse and meaningful visualizations that can answer this question: "What is the elevation range in Nebraska, and how might this affect the stress factors..." - Choose appropriate chart types based on the data. - Group and sort data when relevant especially on the x-axis (e.g., by refArea, refPeriod, or year). - Include meaningful axis labels, chart titles, and legends. - Output only the function wrapped in double quotes (no markdown formatting or explanations).

Table 1
Sample of Prompts Created by the Prompt Builder Component.

5. Conclusion

We presented in this paper our vision of a Generative Knowledge Discovery framework. The proposed framework aims to facilitate the knowledge discovery process through a human-AI collaborative process that dynamically helps to build context and prompts to assist in the generation of questions, queries, and visualizations of information that spans local and external knowledge sources. Our initial demonstration on the bees' context illustrates the feasibility of the approach.

With respect to future research directions, this work can be extended from a theoretical, practical, and technical perspective. At a theoretical level, it is worth exploring the implications of the knowledge discovery process in the context of human-AI collaboration from a socio-technical angle. At a practical level, it may be interesting to explore the impact of this approach in organizational contexts, where knowledge can be locally distributed at departmental and personal levels. At a technical level, it can be valuable to study opportunities for enhancing the framework to automate the identification of relevant external knowledge sources based on the nature of local data. It is also valuable to investigate means for reducing the reliance on user input during the prompt-building stage, and study optimal strategies for user input to implement better and more sophisticated context-building techniques that may lead to more semantically relevant and diverse questions. Furthermore, additional user studies to evaluate the feasibility and usability of the tool in other contexts that include, for example, more generic local knowledge graphs in other domains and schemas beyond RDF Data Cubes and DBpedia. This will inform further design requirements, future research paths, and more potential serendipitous knowledge discovery.

This work offers the following contributions. First, it advances the notion of human-AI collaboration in the context of knowledge discovery on the web of data. Second, it offers new insights into merging the capabilities of knowledge graphs and LLMs to generate insights from data. Third, it provides practical insights through a demo that can be used and extended by researchers and practitioners interested in advancing the field of knowledge discovery at a web-scale.

Acknowledgments

This work was partially supported by the Suliman S. Olayan School of Business (OSB) Research Initiative fund and the American University of Beirut Research Board (URB).

Declaration on Generative AI

During the preparation of this work, the authors used GPT 4.1 and Grammarly (free version) in order to: fix coding, grammar and spelling issues. After using this tool/service, the authors debugged, reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, Knowledge Discovery and Data Mining: Towards a Unifying Framework., in: KDD, volume 96, 1996, pp. 82–88.

[2] N. Yau, Visualize this: the FlowingData guide to design, visualization, and statistics, John Wiley & Sons, 2011.

[3] A. R. Sanders, Data Aggregation and Exploratory Visualization, in: A. R. Sanders (Ed.), Visualizing History's Fragments: A Computational Approach to Humanistic Research, Springer International Publishing, Cham, 2024, pp. 145–185. doi:10.1007/978-3-031-46976-3_5.

[4] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. d. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge Graphs, Synthesis Lectures on Data, Semantics, and

Knowledge 12 (2021) 1–257. doi:10.2200/S01125ED1V01Y202109DSK022, publisher: Morgan & Claypool Publishers.

[5] M. Atzori, G. Koutrika, B. Pes, L. Tanca, Special issue on "Data Exploration in the Web 3.0 Age", Future Generation Computer Systems 112 (2020) 1177–1179. URL: https://www.sciencedirect.com/science/article/pii/S0167739X20324171. doi:10.1016/j.future.2020.07.059.

[6] M. Lissandrini, K. Hose, T. B. Pedersen, Example-driven exploratory analytics over knowledge graphs, in: 26th International Conference on Extending Database Technology, EDBT 2023, Open-Proceedings. org, 2023, pp. 105–117.

[7] A.-C. N. Ngomo, S. Auer, LIMES: a time-efficient approach for large-scale link discovery on the web of data, in: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Three, IJCAI'11, AAAI Press, Barcelona, Catalonia, Spain, 2011, pp. 2312–2317.

[8] J. Volz, C. Bizer, M. Gaedke, G. Kobilarov, Discovering and Maintaining Links on the Web of Data, in: A. Bernstein, D. R. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta, K. Thirunarayan (Eds.), The Semantic Web - ISWC 2009, Springer, Berlin, Heidelberg, 2009, pp. 650–665. doi:10.1007/978-3-642-04930-9_41.

[9] T. Pellissier Tanon, D. Vrandečić, S. Schaffert, T. Steiner, L. Pintscher, From Freebase to Wikidata: The Great Migration, in: Proceedings of the 25th International Conference on World Wide Web, WWW '16, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2016, pp. 1419–1428. URL: https://dl.acm.org/doi/10.1145/2872427.2874809. doi:10.1145/2872427.2874809.

[10] V. Dibia, LIDA: A Tool for Automatic Generation of Grammar-Agnostic Visualizations and Infographics using Large Language Models, in: ACL 2023 (Demonstration track), arXiv, 2023. doi:10.48550/arXiv.2303.02927, arXiv:2303.02927 [cs].

[11] K. Schnizer, S. Mayer, User-Centered AI for Data Exploration: Rethinking GenAI's Role in Visualization, in: arXiv.org, 2025. URL: https://arxiv.org/abs/2504.04253v2.

[12] P. Maddigan, T. Susnjak, Chat2vis: Generating data visualizations via natural language using chatgpt, codex and gpt-3 large language models, IEEE Access 11 (2023) 45181–45193. Publisher: IEEE.

[13] M. R. A. H. Rony, U. Kumar, R. Teucher, L. Kovriguina, J. Lehmann, SGPT: A Generative Approach for SPARQL Query Generation From Natural Language Questions, IEEE Access 10 (2022) 70712–70723. doi:10.1109/ACCESS.2022.3188714, conference Name: IEEE Access.

[14] R. Wang, Z. Zhang, L. Rossetto, F. Ruosch, A. Bernstein, NLQxform: A Language Model-based Question to SPARQL Transformer, 2023. doi:10.48550/arXiv.2311.07588.

[15] C. Peng, F. Xia, M. Naseriparsa, F. Osborne, Knowledge Graphs: Opportunities and Challenges, Artificial Intelligence Review 56 (2023) 13071–13102. doi:10.1007/s10462-023-10465-9.

[16] B. Sarrafzadeh, E. Lank, Improving Exploratory Search Experience through Hierarchical Knowledge Graphs, in: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 145–154. doi:10.1145/3077136.3080829.

[17] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, J.-R. Wen, A Survey of Large Language Models, 2023. doi:10.48550/arXiv.2303.18223.

[18] E. Hyvönen, Serendipitous knowledge discovery on the Web of Wisdom based on searching and explaining interesting relations in knowledge graphs, Journal of Web Semantics 85 (2025) 100852. doi:10.1016/j.websem.2024.100852.