Deliverable from the COST Action CA19134 “Distributed Knowledge Graphs”

Research Agenda

Due date: 30 March 2024

Edited by: Tobias Käfer (DE), Remzi Çelebi (NL), Andreas Harth (DE), Antoine Zimmermann (FR), Ross Horne (LU)

cost
EUROPEAN COOPERATION
IN SCIENCE & TECHNOLOGY

With contributions from:

Anastasia Dimou (BE), András Micsik (HU), Andreas Harth (DE), Andrei Olaru (RO), Antoine Zimmermann (FR), Boris Bikbov (IT), Chang Sun (NL), Giorgos Flouris (GR), Kārlis Čerāns (LV), Katja Hose (DK), Krzysztof Węcel (PL), Michael Freund (DE), Michel Franck (FR), Milos Jovanovik (MK), Nuno Garcia (PT), Olaf Hartig (SE), Remzi Çelebi (NL), Rob Brennan (IE), Ross Horne (LU), Samir Omanović (BA), Sebastian Schmid (DE), Tobias Käfer (DE), Zhen Chen (UK)

This deliverable is based upon work from COST Action "Distributed Knowledge Graphs", supported by COST (European Cooperation in Science and Technology).

COST (European Cooperation in Science and Technology) is a funding agency for research and innovation networks. Our Actions help connect research initiatives across Europe and enable scientists to grow their ideas by sharing them with their peers. This boosts their research, career and innovation.

www.cost.eu

Preface

This deliverable has been compiled by the network of the COST Action "Distributed Knowledge Graphs". This COST Action is a research and innovation network to connect research initiatives around the topic of Distributed Knowledge Graphs. The network met in order to compile research ideas into a research agenda for its field of research.

The ideas have been generated in two virtual sessions in January and February 2024. To spark the ideation process, members of the Core Group defined eight high-level topics, from which the participants of the sessions identified four as the most interesting topics to further discuss. In the weeks that followed the sessions, the topic sponsors condensed the minutes from the sessions into a research agenda for their topic. This phase has been followed by further editing by the action chair.

The topics discussed define the outline of this document:

- LLM-guided Knowledge Graph Generation, Knowledge Discovery, and Knowledge Graph Use
- Representing and Reasoning with Processes in Distributed Knowledge Graphs
- Distributed Knowledge Graphs for Multi-agent Systems on the Web
- Security and Privacy in Distributed Knowledge Graphs

In this deliverable, we present our research agenda under the headlines of those topics.

By sharing this research agenda, the network wants to coordinate its future research, and inform decision makers about potentially high-impact future research directions.

Karlsruhe (DE), Maastricht (NL), Nürnberg (DE), St Étienne (FR), and Luxembourg (LU)

Tobias Käfer
Remzi Celebi
Andreas Harth
Antoine Zimmermann
Ross Horne

LLM-guided Knowledge Graph Generation, Knowledge Discovery, and Knowledge Graph Use

Recent advances in Large Language Models (LLMs) and their potential in processing and structuring unstructured information hint at potentially fruitful combinations with Knowledge Graphs, which are used to model (semi-)structured information. Potential research directions in Knowledge Graphs and Large Language Models, as identified by the network of the COST Action Distributed Knowledge Graphs, include:

1. Generation of Knowledge Graphs and RDF from Structured Data: The ability of LLMs to recognise patterns in their input, combined with the apparent familiarity of LLMs with well-known ontologies hints at the potential of LLMs in Knowledge Graph Creation. Investigations could include (1) to directly map data from various formats such as csv, json, and xml into Knowledge Graphs and Resource Description Framework (RDF), or (2) the LLM-based generation of mappings (e.g. RML mappings) that in turn can be used to convert csv, json, xml to Knowledge Graphs and RDF.

2. Improved Search on and generation of multi-modal Knowledge Graphs with LLMs: If Knowledge Graphs already contain links to other modalities (e.g. images, text), or such links are to be generated, LLMs can potentially be used to generate such links or more of such links. Search techniques that build on vector-based semantic similarities, e.g. to build retrieval-augmented generation (RAG) systems, may also benefit from the combined processing of graph embeddings and embeddings based on other modalities.

3. Automated Ontology Engineering: As part of the development of ontologies for a given application domain, LLMs could get employed to automatically generate a taxonomy in this domain, maybe even with connections between the terms in the taxonomy, or to assist in the generation of this ontology. We discussed the automatic construction of Knowledge Graphs for different domains, focusing on tasks such as populating KGs for already defined underlying ontologies.

4. Knowledge Graphs as part of Agentic LLM-based workflows: Research is required to bring the benefits of Knowledge Graphs to LLM-powered systems. Specifically, queries to Knowledge Graphs can be answered fastly and hallucination-free by an easily updatable Knowledge Graph. Such querying systems need to become part of an agentic LLM-based system and correspondingly orchestrated.

5. Factual Knowledge Verification and Information Extraction: LLMs could help in verifying factual knowledge in Knowledge Graphs, e.g. by identifying relevant documents using which this knowledge can be substantiated. Conversely, the output of LLMs, which may or may not include so-called hallucinations, can be processed using entity recognition and relation extraction techniques or embeddings, which again may be executed using LLMs, to double-check those relations using Knowledge Graphs.

Representing and Reasoning with Processes in Distributed Knowledge Graphs

Knowledge Graphs use standard technologies such as URIs, RDF, and ontologies to represent factual knowledge on the Web. However, the modelling of certain applications, ranging from risk governance systems and digital twins in manufacturing to workflow provenance and agent-based collaboration scenarios, requires the modelling of sequential steps to achieve goals, which is currently not possible. To enable the modelling of such scenarios, which in essence model a process, a standardised, Web-first approach to process representation is required.

To achieve such a representation, a collaborative research effort is needed to develop a standardised, Web-first process formalisation that integrates seamlessly with Knowledge Graphs focused on creating a universally adopted standard across multiple industries. Required research areas identified by the members of the COST Action for Distributed Knowledge Graphs are:

1. Event log formalism: Formulating a method for representing event logs and mapping between different formats is essential. This includes ensuring immutability of logs, possibly through blockchain technology.

2. Human-centred process modelling: Expanding standardisation research to include human-centred processes, such as emergency room medical procedures, in addition to software or industrial processes.

3. API discovery and design: A challenge is to design APIs that enable complex, agent-based process orchestration tasks without extensive documentation.

4. Neuro-symbolic approaches: Combining symbolic process representation with machine learning, especially LLM, can be a promising research direction.

5. Process Mining and Knowledge Graph synergy: Combining Process Mining with Knowledge Graph based process representations such as Workflow provenance could provide reasoning-based insights for process optimization.

6. Improving Computer Vision with Process Knowledge: Explore the potential of process knowledge to improve computer vision tasks by identifying pitfalls in the process, leading to more informed and accurate data interpretations.

Distributed Knowledge Graphs for Multi-agent Systems on the Web

Multi-agent systems are systems composed of autonomous entities that can observe their environment, communicate with other such entities, and take decisions either individually or collectively. It is typically assumed that agents have knowledge of their own encoded formally in some language that allows them to take a meaningful decision in function of their goals and the situations they observe. This knowledge may be based on Knowledge Graphs or the agent may have to interact with an environment described in Knowledge Graphs. It is possible to facilitate the decision making process by presenting to the agent a structured description of a piece of the environment that matches the agent's internal ontology. For instance, on the Web, agents can autonomously perform actions over web resources if there is a formal representation of the resource affordances, making explicit what is possible to do with a web platform, how it is possible to do it, and possibly the outcome of doing it. As an example, an agent may have a general ontology of web forum interaction (reading and writing posts, replying to posts, starting new threads, etc.) and only requires a concise and structured description of a message board in order to participate in discussion autonomously. To fully take advantage of web resources, an agent must consume (and may produce) distributed knowledge that is interlinked.

Developing Distributed Knowledge Graph-driven multi-agent systems requires a large research programme to address many issues, as identified by the network of the COST Action Distributed Knowledge Graphs:

1. Knowledge engineering: agents must have a formal representation of their own knowledge or beliefs, but there must be ontologies that serve representing parts of the environment. An agent working in a physical environment may have a map or 3D representation of the space, but it may also have access to a higher level description of how the space is organised (e.g., this zone is a delivery area). A virtual or online agent may have to know the intended function of a web service, etc.

2. Resource description discovery: In order to navigate arbitrary environments without the need for infinite knowledge, agents should discover knowledge as they move (physically or virtually) and access self-describing resources. In an online environment, the navigation can be supported by hypermedia, and the knowledge discovery supported by Linked Data, following and dereferencing links. But there must be research done on the architecture for publishing, managing and storing Distributed Knowledge Graphs so that agents know where to look for certain kinds of knowledge.

3. Knowledge Graph-based interaction: Any interactions can be based on reading or writing pieces of knowledge graphs. There are agent-environment interactions where agents consume Knowledge Graphs discovered in the environment and produce content and traces as Knowledge Graphs that other agents may have access to. Agent-agent interactions may also rely on exchanging Knowledge Graphs. Protocols may have to be devised, possibly with vocabularies dedicated to the interaction dimension.

4. Coordination: coordination among agents may be supported by specific vocabularies describing the intended organisation of a workspace (e.g., making explicit the roles in relation to goals and tasks). This challenge may involve yet more knowledge engineering to design a coordination ontology, but also protocols that define how agents can take advantage of the knowledge of the organisation of the space to adopt appropriate collaborative behaviours.

5. Regulation: regulation can be supported by explicit policies and norms that agents can manipulate, reason with, and follow. Work on regulation languages and how to encode them are necessary, and principles for how to enforce them are crucial.

Security and Privacy in Distributed Knowledge Graphs

Security and privacy problems related to distributed knowledge graphs can be discussed on several levels:

1. At the most basic level, graphs may express knowledge that should be treated confidentially and hence should not flow freely through a distributed system. For example, personal data about patients in the medical domain is subject to processing restrictions governed by GDPR.
2. There are also knowledge graphs that should satisfy integrity properties that ensure that the contributors to the data did adhere to adequate professional practices when producing or processing data before it is published in a given context, requiring trusted explanations of the origin of data used in industrial decision making.
3. Ensuring security policy models are supported by systems to provide adequate guarantees is both a human and technical problem which is unavoidable for regulatory reasons.
4. Besides knowledge graphs themselves satisfying confidentiality or integrity properties mentioned above, there is also the problem of how knowledge graphs themselves could be used in the design and implementation of security policy models for distributed systems with confidentiality and integrity requirements.

Correspondingly, the network of the COST Action Distributed Knowledge Graphs identified four pillars for future research directions:

1. Knowledge graphs for access. There are several vocabularies for specifying access control policies notably Web Access Control (WAC) and Access Control Policy (ACP). They can all express a policy where users can perform specific operations on data. What they do not provide is a security policy model that ensures that security policies are used such that specifically chosen confidentiality and integrity properties are maintained, such as information intended for a user logged into one app should not flow to another app when the same user log into it. A related problem is that when database backends are exposed as knowledge graphs then the access control information exposed for the knowledge graph ideally should respect the integrity and confidentiality properties of the database, but, in reality, the database design pattern of sanitising data for a specific purpose and then placing access policies on the sanitised data is more feasible. Criteria for determining whether sanitisation is adequate with respect to a security policy model could be devised. Knowledge graph schemas and ontologies can also augment WAC, ACP etc., for describing patterns of access such as granting access to a SHACL shape within a dataset.

2. Knowledge graphs for compliance. Ideal policy models should not only control access but also should make explicit legal and ethical permissions and obligations. For example, when access is granted to personal data the Controller associated with the request should be identified as part of the policy. Furthermore, information such as the contact details of the Controller are obliged to be recorded under GDPR. Obligations such as retention periods

may not be enforceable technically but can be regulated legally. Concrete steps have been taken to provide suitable legal vocabularies, using ODRL for instance, for expressing such information as knowledge graphs. Significant work is still required to devise policy models suitable for ecosystems such as Solid. People should be accounted for in the creating and evaluation of policies.

3. Knowledge graphs for authentication. The W3C Verifiable Credential standard allows knowledge graphs to be signed cryptographically with the information about the signature also forming a knowledge graph. The primary use case for verifiable credentials is that a user's credentials are signed by an issuing authority, and then the credential is signed again by the user to approve its usage in a specific context. Zero-knowledge proofs may also be supported for increased privacy, but they rely on stronger trust assumptions about a group of credential holders. Verifiable credentials create layers of knowledge graphs about knowledge graphs whose trust is supported by public key infrastructure which may also use knowledge graph technology such as DID documents asserting public keys. There is an opportunity for better convergence and interoperability by relating VCs to HTTPSig and DPoP tokens, for example. Policies are enforced by matching against appropriately authenticated information, e.g., informed consent should be agreed between parties cryptographically, where agreement is an authentication property of a well-designed protocol.

4. Protocols. Components of distributed systems are connected in a distributed system. Authentication is achieved by the correct usage of VCs within an appropriate protocol. Knowledge graphs containing confidential information or for coordinated decision making between apps of multi-agent systems (c.f., FIPA, JIAK framework) are delivered via protocols. Existing protocols used in knowledge ecosystems have been found to contain vulnerabilities and standards have been found to be underspecified in such a way that man-in-the-middle attackers can elevate their privileges for example. There is a need to make explicit the threat models with respect to which protocols can be certified as secure. Even from the perspective of security research, systems involving distributed knowledge graphs are non-trivial to design and verify since they typically rely on multiple distributed entities for storing data, processing data, and trust management.

We could ask why basic security mistakes are made when designing distributed knowledge graphs. However, more positively we may rather ask: "How can we make it obvious to developers how to design distributed systems incorporating knowledge graphs that are adequately secure and private."