Narrative Understanding with Knowledge Graphs

Blin, Inès Edith Laurent

2026

DOI (link to publisher)
10.5463/thesis.1517

document version
Publisher's PDF, also known as Version of record

NARRATIVE UNDERSTANDING WITH KNOWLEDGE GRAPHS

This work was partly funded by the European MUHAI project, grant no. 951846, by the XS NWO Project, grant no. 406.XS.04.118, and by the Sony Computer Science Laboratories - Paris.

SIKS Dissertation Series No. 2026-07
The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

DOI: https://doi.org/10.5463/thesis.1517
Typeset by LaTeX.
Printed by: Ridderprint, the Netherlands
Cover design by: Ridderprint, the Netherlands

VRIJE UNIVERSITEIT

NARRATIVE UNDERSTANDING WITH KNOWLEDGE GRAPHS

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad Doctor of Philosophy aan
de Vrije Universiteit Amsterdam,
op gezag van de rector magnificus
prof.dr. J.J.G. Geurts,
volgens besluit van de decaan
van de Faculteit der Bètawetenschappen
in het openbaar te verdedigen
op donderdag 5 februari 2026 om 11.45 uur
in de universiteit

door

Inès Edith Laurent Blin

geboren te Parijs, Frankrijk

promotor:	prof.dr. A.C.M. ten Teije
copromotoren:	dr. I. Tiddi dr. R. van Trijp
promotiecommissie:	prof.dr. J.R. van Ossenbruggen dr. V. Presutti prof.dr. E. Motta dr. M.G.J. van Erp prof.dr. P.T.J.M. Vossen

Acknowledgements

This dissertation is not only the result of serendipitous encounters within different research institutes, but also of many enriching discussions with talented people. Above all, it grew out of a fruitful collaboration between my two research environments, the Sony Computer Science Laboratories in Paris, and the Vrije Universiteit Amsterdam, and through the MUHAI EU project, where everything began. It also reflects the many journeys I undertook, mostly between Paris and Amsterdam, but also across Europe and beyond.

I would like to first thank my supervisors Annette ten Teije, Ilaria Tiddi, and Remi van Trijp for their continuous support and guidance throughout my PhD. I am deeply grateful to all three of them for the trust they placed in me, the freedom they gave me to pursue my ideas, and their constant availability whenever I needed advice. From Annette, I learned to maintain a broader vision, to stay focused on research questions that truly matter, and to move forward without getting lost along the way. With Ilaria, I greatly valued our continuous brainstorming, her help in structuring my ideas more clearly, her faith in my work and her motivational speeches. I am also grateful for the time she took to secure the NWO project and contract when I needed additional funding. Thank you, Remi, for encouraging me to explore a new research direction in the lab, one that was previously uncharted, and for being supportive in all circumstances. I particularly appreciated our discussions, which helped me better articulate and develop the bigger vision of my research.

Although not formally my supervisor, I would like to thank Frank van Harmelen for often acting as one in practice. Our discussions, sometimes in the spirit of the *"rubber duck method"* were invaluable in helping me test ideas, uncover their potential, and identify their weaknesses. I was also inspired by

the way he led the lab in Amsterdam and by the cohesion he fostered within it.

Thanks Annette, Ilaria, Remi and Frank for giving me the opportunity to pursue this PhD in this particular setting, and for welcoming me into your labs.

I would like to thank Giuliana Spadaro for hiring me for the NWO project and for our many discussions exploring the intersection of social sciences and AI, as well as for her enthusiasm, availability, and positive energy. I also thank Daniel Balliet for valuable discussions during that time.

I had the fortune of spending the majority of my thesis in the Sony Computer Science Laboratories in Paris. I am deeply grateful to Martina Galletti for her continuous support throughout our PhDs, in both good and challenging times, whether in or out of research. Thank you for your energy in starting new projects, in connecting people, and for bringing such a joyful presence to the lab. I would like to thank Giulio Prevedello for all our discussions, especially the ones on statistics and brainstorming ideas, but also out of research. To both of you, I appreciated all the coffees we shared, which made working together even more enjoyable, and I learnt a lot from our fruitful collaborations. Thank you for being such good friends. I am also grateful to Peter Hanappe for the way he encourages us to explain our research, as well as for the inspiring and welcoming way he leads the lab. I thank David Colliaux for his renewed interest in my work and the many valuable discussions we had. With Cyran Aouameur and Cristina Nunu, I cherish all the fulfilling moments in the lab or on the tennis court. Lastly, I would like to thank Aliénor Lahlou for introducing me to the lab in the first place.

I also want to thank other members (current and former) of CSL, which in one way or another I had the pleasure to interact with: Alexis André, Michael Anslow, Théis Bazin, Pratik Bhoir, Doug Boari, Matteo Bruno, Milena Di Canio, Samir Dhane, Amaury Delort, Matthias Demoucron, Emmanuel Deruty, Elisabetta Falivene, Nidham Gazagnadou, William Gaultier, Pietro Gravino, Ben Hayes, Gabriel Hurtado, Zineb Lahrichi, Stefan Lattner, Alessandro Londei, Vittorio Loreto, Sébastien Marino, Lenon Modesto, Alida Ndayakire, Javier Nistal, Alfred Pichard, Alain Riou, Ruggiero Lo Sardo, Nabil Ait Talib, and Michael Turbot. From Sony Europe, I thank Florence Baylin, Coraline Cadran and Catherine Delamare for their support.

At the same time, I am grateful to the Vrije Universiteit Amsterdam, where I was officially enrolled as a PhD student and had the opportunity to visit regularly. Although we did not collaborate on a project in the end, I am thankful to Márk Adamik for all the stimulating discussions we had. I am grateful to Nikos Kondylidis for our initial discussions that convinced me that his lab would be a great place to work. Thank you both for the continuous conversations that helped us push through our PhDs, and for the many enjoyable moments we

spent together. I also thank Lise Stork for her mentoring, our discussions and collaborations, from which I learnt a lot about effective teamwork in writing and coding. I am grateful to Taraneh Younesian, Daniel Daza (thanks for the template!), and Michael Cochez for their expertise and availability during many valuable discussions. I am thankful for the time spent with Taraneh, Jan-Christoph Kalo, Ritten Roothaert, Romy Vos and Giacomo Zamprogno, whether in the usual "PhD" room, during board (or was it card?) games, or else in countless other moments, which made the lab experience memorable. I would like to thank Floris den Hengst, Majid Mohammadi, Ihtesham Shah, and Taraneh for an enriching collaboration that started with an outing and grew into a concrete outcome. Finally, I thank Mojca Lovrencak for her support and availability throughout the PhD.

I also want to thank other members (current and former) of the Knowledge Representation and Reasoning (KRR), Learning and Reasoning (L&R) and Knowledge in AI (KAI) groups, which in one way or another I had the pleasure to interact with: Erman Acar, Dimitrios Alivanistos, Selene Baez Santamaria, Ruud van Bakel, Peter Bloem, Gabriella Bollici, Yannick Brunink, Jieying Chen, Chaohui Guo, Loan Ho, Fabian Hoppe, Filip Ilievski, Atefeh Keshavarzi Zafarghandi, Ali Khalili, Mayank Kharbanda, Hossein Khojasteh, Taewoon Kim, Patrick Koopmann, Emile van Krieken, Benno Kruit, Margherita Martonara, Ameneh Naghdipour, Romana Pernisch, Andrea Rafanelli, Andreas Sauter, Stefan Schlobach, Thibaut Soulard, Thiviyan Thanapalasingam, Shuai Wang, Xu Wang, and Xander Wilcke.

Lastly, this work was enriched by stimulating collaborations and discussions within the MUHAI EU project, which provided a broader context. I am particularly grateful to Lise, Laura Spillner and Carlo Santagiustina for our collaboration that began as a "nano-project" and grew into a fully realized outcome. I would also like to thank all other project members not mentioned so far for the many insightful discussions, whether online or during the numerous project meetings: Katrien Beuls, Susanna Cappello, Alessandra Fornetti, Rainer Malaka, Ilda Mannino, Davide Michielin, Anna Morbiato, Jens Nevens, Mihai Pomarlan, Robert Porzel, Luc Steels, Paul Van Eecke, Lara Verheyen, Oscar Vilarroya, Gerald Volkmann, Pieter Wellens, and Folco Soffietti.

To finish, I would like to truly thank my partner Emmanuel for his unwavering support throughout this journey, during both good times and more challenging moments. Thank you for believing in me and valuing my research, for your genuine interest in my work and for encouraging me to discuss about it. Finally, I thank my parents, family, and friends for their constant support and encouragement.

Many figures in this thesis used resources from Flaticon.com.

Contents

1 Introduction

Telling stories and constructing narratives is a fundamental human trait [42, 46, 136]. Narratives help us make sense of experience, explain events, and establish shared understanding [42, 341]. These abilities come naturally: humans intuitively identify key entities, infer relationships, and connect events into coherent interpretations [42]. Building computational systems that emulate these narrative capabilities can make AI more human-centric, that is better aligned with how people reason, explain, and communicate [250, 310].

While narratives are often associated with textual storytelling, the underlying process goes beyond surface language. Constructing a narrative involves retrieving relevant events, identifying causal or thematic links, and organizing them into meaningful structures. As such, narratives serve a broad range of cognitive functions, from explaining and predicting to generating hypotheses and analysing discourse in complex domains. Narratives therefore play a central role in understanding, conceived as the construction of meaningful models that link new observations to background knowledge and support tasks such as answering questions, explaining outcomes, planning, and predicting [310].

The study of narratives in AI has its roots in narratology, that investigates narrative structures, elements, and interpretation [20, 155, 166]. Early AI systems drew on these foundations to build symbolic, machine-readable representations that captured events, roles, and causal links. These structured representations were often used for tasks such as narrative generation, retrieval, and comparison, and were typically grounded in social and cultural contexts, as consistently depicted in the *Computational Models of Narrative* workshop series [111–114]. In recent years, the field has shifted toward data-driven approaches, particularly with the rise of deep learning [293]. Large language models (LLMs) now tackle tasks like narrative generation and coherence di-

rectly from raw text. This is evidenced by the rising number of workshops such as *the Workshop on Narrative Understanding, Text2Story, the Workshop on Computational Creativity in Language Generation, Events and Stories in the News*, or *Computing News Storylines*. In parallel, knowledge-based approaches have remained prominent in domains such as digital humanities and biomedicine, where curated knowledge graphs (KGs) enable reasoning, discovery of connection, and the reconstruction of storylines [73, 82, 322].

While LLMs such as ChatGPT [254] have demonstrated remarkable proficiency in generating fluent and contextually appropriate text [5, 197], they still face well-documented limitations in transparency, verifiability, and structured reasoning [284]. In narrative sensemaking tasks where coherence, causal logic, and evidential grounding are essential, these limitations become particularly consequential. LLMs can struggle with factual consistency, may hallucinate content [178], and often lack mechanisms for tracing how conclusions are derived. Moreover, LLMs are primarily trained to generate surface-level linguistic output, whereas narratives may assume diverse structural forms that go beyond textual fluency, such as causal chains, thematic arcs, or graph-based explanations. As such, relying solely on LLMs may be inadequate for high-stakes applications that demand interpretability, accountability, and trustworthiness. On the other hand, knowledge-based approaches, and KGs in particular, offer interpretability and structured reasoning. They also present notable challenges such as data curation, accuracy and consistency, or the integration of heterogeneous data sources [161, 262], although current research is alleviating some of these burdens.

This thesis builds on both traditions, combining the interpretability and structure of symbolic representations with the adaptability of data-driven methods, to support narrative construction across heterogeneous domains. This thesis adopts a working understanding of narrative as a representation, often textual, derived from underlying entities and relationships captured in knowledge graphs, that helps make sense of observations in a given domain. This thesis investigates automated narrative construction through hybrid, knowledge-enriched approaches. I argue that structured, machine-actionable representations grounded in events and relationships can more effectively support narrative functions such as explanation, hypothesis generation, and discourse analysis, offering greater interpretability and robustness. KGs, as a form of structured representation, bring key advantages for organizing and reasoning over narrative content. By explicitly modeling entities and their relationships, KGs provide a natural foundation for structuring memory and contextual knowledge. Their formal structure supports data lineage, transparency, and explainability [201], qualities increasingly prioritized in AI research. In

narrative applications, KGs enable the construction of rich, interpretable representations that enhance both comprehension and generation [262]. KGs offer semantic grounding and traceability, while neural approaches contribute fluency and adaptability. Rather than treating structured approaches and neural approaches as mutually exclusive, this thesis considers them as complementary, as also suggested in the literature [127].

As AI systems are increasingly applied in socially and historically sensitive domains, it becomes essential to develop representations that support transparent, structured, and interpretable narratives. Several AI systems have been proposed to assist users in constructing such narratives, such as retrieving and organizing news articles for event-centered storytelling [342], connecting relevant events in legal cases to support argumentation [110], or navigating archival material to generate historical hypotheses [238]. This thesis builds on that motivation through a hybrid strategy combining symbolic and statistical representations. This methodology forms the core of the work and is applied across diverse domains: historical question answering, social media discourse analysis, and hypothesis generation in the social sciences, further detailed in Sections 1.2 and 1.3 of this chapter.

1.1 Research Questions

How can knowledge graphs enable narrative-like functions such as explanation, hypothesis, and discourse analysis? This is the issue that is addressed in this thesis, and that is explored through three research questions.

Research Question 1 (Representation): How to *represent* narrative-like structures as knowledge graphs?

Although narratives play a central role in how humans explain, interpret, and communicate complex phenomena, few computational approaches have systematically identified their core structural elements, such as events, agents, and causal links, or unified existing representation models. Chapter 2 addresses this gap by analyzing the representational requirements for narrative construction and investigating how existing ontologies can fulfill them. Based on expert interviews, this thesis proposes a unified terminology and offers practical guidelines for selecting ontological models suited to different narrative use cases. These guidelines are then applied across the three domains explored in this thesis: modeling historical narratives (Chapters 3 and 4), querying event structures in social media data (Chapter 5), and formalizing social science hypotheses through expert-driven design (Chapter 6).

Research Question 2 (Construction): How can narrative-centric knowledge graphs be *constructed* from heterogeneous data sources across domains?

Constructing structured, narrative-centric representations involves two core steps: retrieving relevant content and transforming it into KG form. Chapter 3 introduces a semantically guided retrieval method for identifying event-relevant subgraphs. Domain-adapted KG construction strategies are then applied across the three use cases: historical data (Chapters 3 and 4), social media content (Chapter 5), and social science hypotheses (Chapter 6).

Research Question 3 (Evaluation): How *effective* are structured narrative representations for supporting reasoning tasks, such as question answering or hypothesis generation?

Evaluating structured narrative representations requires distinguishing between intrinsic, quantitative performance and extrinsic, qualitative value. Following a task-based approach common in the literature, this thesis assesses the value of hybrid approaches both through metrics from standard tasks (e.g., link prediction, classification) and through user studies that reflect real-world reasoning tasks. Chapter 3 combines a metric-based evaluation with a qualitative study comparing the outputs of large language models prompted with and without structured knowledge. Chapter 4 investigates the effect of narrative structure, both syntactic and semantic, on model performance. Chapter 6 extends this evaluation to hypothesis generation in a social science setting, where domain experts assess and compare AI-generated hypotheses with those they developed themselves.

1.2 Approach

To explore how narratives can be operationalized in AI systems, this thesis proposes a general methodology that guides the construction and use of structured, machine-actionable representations. Narratives rely on common cognitive processes: identifying meaningful connections, and assembling coherent representations that support explanation, hypothesis generation, or discourse analysis. These processes are grounded in memory, which may be personal (reflecting individual knowledge and experience) or collective (drawing from shared or domain-specific knowledge). The proposed method captures these shared components in a flexible but consistent workflow, structured around three main steps that enable the systematic and scalable generation of narratives, and that fits the three research questions identified in Section 1.1. Figure 1.1 provides an overview of our approach.

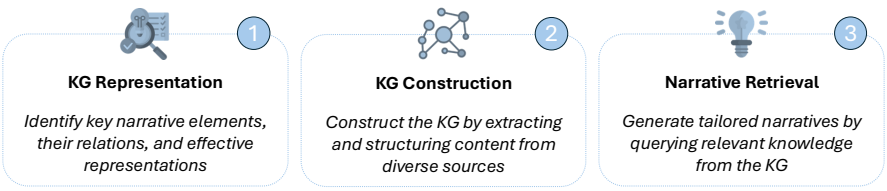

Figure 1.1: Overview of the proposed approach.

1. **KG Representation.** This symbolic step determines the representational requirements by identifying the key elements and relations that must be captured from narratives. It is then possible to select or define an appropriate ontology to support structured representation.

2. **KG Construction.** This step builds the KG that acts as a structured memory for narrative generation. It extracts relevant content from diverse sources (e.g. text, structured data), structuring the content into a graph aligned with the chosen ontology, and validating the graph's consistency. This step is primarily symbolic, though it may incorporate neural-based methods for information extraction.

3. **Narrative Retrieval.** Once the KG is constructed, it serves as a dynamic foundation for generating varied narratives. Specific KG queries select relevant subsets of knowledge, which are then structured or assembled into narratives tailored to the intended function, be it explanation, hypothesis generation, or discourse analysis. The flexibility of the KG allows generating different narratives from the same underlying KG, enabling personalized or contextualized storytelling with lineage. This step is primarily neural, though it may incorporate symbolic methods for extracting the information needed from the KG.

While the technical implementations vary depending on domain-specific input, requirements and constraints, this approach offers a conceptual and methodological scaffold.

1.3 Application Domains

To demonstrate the applicability of our approach for structured narrative construction, we selected three use cases spanning a wide spectrum of input data and KG types, temporal orientations, and narrative functions:

Table 1.1: Summary of use case characteristics.

Attribute	Historical	Social Media	Social Science
Input data	KG, Text	Tabular data, Text, Metadata	KG
Memory	Generic	Personal	Domain-specific
Type	Compression	Compression	Compression, Expansion
Methods	Symbolic, LLM	Symbolic	Symbolic, Neural
Narrative	QA	Belief Narratives	Hypothesis Generation

- **Historical Data:** Focused on event-centric narratives grounded in notable historical events, this use case utilizes KGs and text as input to support narrative explanations and question answering. It reflects a shared memory type and emphasizes compression of vast historical knowledge.
- **Social Media Discourse:** Centered on real-time, personal data including tabular data, text, and metadata, this use case highlights narratives about beliefs and framing in ongoing conversations. It corresponds to a personal memory type and involves narrative compression.
- **Social Science Research:** This domain, supported by expert-curated KGs, enables hypothesis generation and reflects a domain-specific memory type. It requires collaboration with specialists to validate knowledge and narrative templates, and involves narrative compression (summarizing insights from past cases), and narrative expansion (generalizing these insights to new data).

These cases were chosen to cover a broad range of practical challenges: from compressing historical knowledge for explanation, to observing social dynamics in discourse, to generating and testing hypotheses in scientific research. Table 1.1 summarizes their characteristics. The table also illustrates how the working understanding of narrative adopted in this thesis manifests across use cases: in historical data, narratives support question answering grounded in notable events, which can be characterised using formal representations such as the SEM [338] ontology, in social media, narratives capture beliefs and interpretations about events, where event boundaries are less formalised, and in social science research, narratives take the form of hypotheses synthesizing findings across studies, providing explanations about patterns of human cooperation and group behaviour. Although each use case in this thesis involves different data formats, domain assumptions, and implementation

choices, they are united by the shared high-level approach described in this section. They illustrate the broad applicability of our approach and demonstrate how it can be adapted and instantiated to meet the specific demands of different data environment and narrative objectives.

1.4 Thesis Layout

This thesis first investigates how narrative requirements can be represented in AI systems. In particular, Chapter 2 examines how structured machine-readable representations such as KGs can model core narrative elements.

The approach described in Section 1.2 is then applied across the three distinct use cases from Section 1.3. Chapters 3 and 4 focus on historical data, demonstrating how structured representations can support question answering, explanation and link prediction. Chapter 5 explores the social media domain, where narrative elements are extracted from unstructured discourse. Lastly, Chapter 6 examines applications in the social sciences, highlighting how structured representations can support hypothesis generation and aggregation.

1.5 Publications

This thesis is based on several publications aimed at answering the previous research questions. These publications form the content for Chapters 2 until Chapter 6, as described in the following section together with my personal contributions. For all publications co-authored by Giuliana Spadaro, dr. Spadaro was responsible for funding acquisition, supported by the NWO XS Project, grant no. 406.XS.04.118. For all other publications, dr. van Trijp held the funding acquisition role, supported by the MUHAI project, grant no. 951846 and by the Sony Computer Science Laboratories - Paris. All computing resources were provided by the Sony Computer Science Laboratories - Paris. Unless otherwise specified, other authors in these works had important advisory and supervision roles and were involved in the review and editing of the manuscripts.

Chapter 2 is based on the paper:

- Inès Blin, Annette ten Teije, Frank van Harmelen, and Ilaria Tiddi. "Structured Representations for Narratives". In: *The 24th International Conference on Knowledge Engineering and Knowledge Management*. 2024.

I carried out the main analysis and the literature review in the paper, produced the visual materials, and wrote and finalized the manuscript. The idea

for the paper originated with Annette ten Teije and Frank van Harmelen, who also conducted the interviews featured in the paper.

Chapter 3 is based on the paper:

- Inès Blin, Ilaria Tiddi, Remi van Trijp and Annette ten Teije. "Chrono-Grapher: Event-centric Knowledge Graph Construction via Informed Graph Traversal". In: *Semantic Web Journal* (2025).

I initiated and developed the main research idea, gathered and prepared the data, designed and implemented the methodology, and carried out the investigation and analysis. I also developed the software tools, validated the results, created the visual materials, and wrote and finalized the manuscript.

Chapter 4 is based on the paper:

- Inès Blin, Ilaria Tiddi and Annette ten Teije. "Measuring the Impact of Narrative Complexity on Knowledge Graph Embeddings". In: *The 24th International Semantic Web Conference*. 2025.

I conceived the research idea, collected and organized the data, designed the methodology, and conducted the investigation and analysis. I also implemented the software components, verified the results, created the visualizations, and wrote and finalized the manuscript.

Chapter 5 is based on the paper:

- Inès Blin, Lise Stork, Laura Spillner, and Carlo Santagiustina. "OKG: A Knowledge Graph for Fine-grained Understanding of Social Media Discourse on Inequality". In: *The 12th Knowledge Capture Conference*. 2023.

The paper was jointly conceived by all four authors during a meeting, where the overall design of the KG, ontology, metadata selection, analysis tools, and data mapping was initiated. The idea of using a KG as a back-end database was initially proposed by Carlo Santagiustina and Laura Spillner. I led the core analysis, developed most of the new software components for constructing the KG, coordinated between the authors, and created the visual materials. Together with Lise Stork, I contributed to validating the data and drafting the initial manuscript. Carlo Santagiustina and Laura Spillner contributed analyses on grammar, sentiment, and entity matching. Carlo Santagiustina also collected the Twitter data and helped frame the socio-economic context and use case.

Chapter 6 is based on the paper:

- Inès Blin, Ilaria Tiddi, Giuliana Spadaro, and Annette ten Teije. "Automated Hypothesis Generation for Human Cooperation Studies". In: *The 4th International Conference Series on Hybrid Human-Artificial Intelligence*. 2025.

I initiated and shaped the research idea, collected and prepared the data, designed the methodology, conducted the investigation and analysis, and developed the supporting software. I also validated the results, produced the visualizations, and wrote and finalized the manuscript.

The followings are additional publications in which I contributed that are relevant to the research topics in this thesis:

- Inès Blin. "Building Narrative Structures from Knowledge Graphs". In: *The 19th European Semantic Web Conference (Satellite)*. 2022.
- Inès Blin. "Building a French Revolution Narrative from Wikidata". In: *Semantic Techniques for Narrative-Based Understanding @IJCAI-ECAI*. 2022.
- Inès Blin, Ilaria Tiddi, Giuliana Spadaro, and Annette ten Teije. "Living Meta-Review Generation for Social Scientists: An Interface and A Case Study on Human Cooperation". In: *The 24th International Conference on Knowledge Engineering and Knowledge Management (Posters & Demos)*. 2024.

2 Structured Narrative Representations

Based on [31]:
Inès Blin, Annette ten Teije, Frank van Harmelen and Ilaria Tiddi
Structured Representations for Narratives
International Conference on Knowledge Engineering and Knowledge Management 2024

While neural methods excel in linguistic fluency and text generation, they often lack the formal structure required for deeper reasoning. Knowledge-based approaches offer semantic clarity and interpretability, yet struggle with the flexibility of natural language. In this chapter, we focus on how knowledge graphs can be used to model key narrative elements. We propose guidelines for selecting appropriate representations based on specific use cases.

Abstract. Narratives are essential to the human sense-making process, and have sparked interest across multidisciplinary fields including Computer Science. However, efforts to unify terminology and identify core narrative requirements are limited, complicating the choice of a suitable representation. We first map identified narrative requirements to appropriate knowledge representations (1). We identify and organise narrative requirements from interviews we conducted (1a), and map these requirements to their most suited knowledge representation forms (1b). We then conduct a systematic survey to select candidate event and narrative-centric ontologies (2). We analyse how effectively these ontologies represent the main narrative requirements (2a), and rank them based on a five-star rating system (2b). We lastly provide recommendations on how to choose the best ontology for a narrative for a specific use case (3).

2.1 Introduction

Narratives are fundamental for human understanding [46], offering invaluable insights into intricate events. They provide coherence to otherwise fragmented and unrelated data, building up coherent explanations from past data. Systems capable of constructing such narratives would be beneficial in developing truly "human-centric" systems. Such systems have already proved valuable in helping building daily narratives, e.g. in the legal [110] and historical [238] domains.

Throughout the literature, the term "narrative" is widely used but with varying interpretations and expectations, drawing from fields such as linguistics, philosophy, narratology, and artificial intelligence [56]. Narrative construction has gained prominence within Computer Science, as evidenced by workshops like the Computational Models for Narratives Workshop Series (2009-2016, [240]), the Narrative Understanding Workshop Series[1] (2019-2023), the Text2Story[2] (2018-2024), or the SEMMES[3] (2023-2024) workshops. Despite the importance of narratives, few efforts have been made to identify core elements common to each work such as events and agents, and to unify existing representations. A recurring simplification defines a narrative as a sequence of events.

The primary objective of this work is to identify the most suitable method, given a set of requirements, for representing a narrative in the form of a knowledge graph (KG) where an ontology is populated with data, that we call *narrative graphs*. The benefit of representing narratives in a structured format like KGs lies in their incorporation of a semantic layer allowing finer-grained representations. Narrative graphs, designed to mirror how humans build and understand narratives [46], facilitate understanding via improved querying and reasoning. KGs have been used as background knowledge to support narrative construction [139]. Debates persist at the syntactic level about optimal KG data models [316], encompassing options like simple RDF, Named Graphs[4] or RDF-star[5]. At the semantic level, the increasing number of narrative resources (ontologies, KGs, etc) make it challenging to choose a suitable narrative representation for a given set of requirements or a use case.

In this paper, we propose to unify the terminology across the literature and to provide guidelines on which ontology model best to use for a specific use case. We divide our problem into three research questions:

[1] `https://sites.google.com/umass.edu/wnu2023/home`
[2] `https://text2story24.inesctec.pt`
[3] `https://anr-kflow.github.io/semmes/`
[4] `https://www.w3.org/2009/07/NamedGraph.html`
[5] `https://w3c.github.io/rdf-star/cg-spec/editors_draft.html`

RQ1: What are the common requirements related to narratives?

RQ2: Can existing ontologies represent these narrative requirements?

RQ3: How to best choose a structured representation for a narrative?

To answer **RQ1**, we conduct interviews to identify narrative requirements, and we organise these requirements into themes. We also map these requirements to their most suited knowledge representation forms. To answer **RQ2**, we conduct a systematic review on event-centric and narrative-centric ontologies, and evaluate the ontologies based on how effectively they can fullfill narrative requirements and on their vocabulary usage. To answer **RQ3**, we lastly provide recommendations on how to best identify a structured representation for a narrative, and apply them to the interviews use cases.

2.2 Narrative Requirements

We elicited narrative requirements from interviews to answer **RQ1**, and we organise these requirements into thematics. We also map these requirements to their most suited knowledge representation (KR) forms.

2.2.1 Identification of Narrative Requirements

We conducted 6 narrative interviews with 9 participants in the MUHAI[6] EU research consortium. MUHAI aims to advance AI systems by integrating narratives to enhance meaning and understanding. The project has two main use cases: pragmatic everyday knowledge (e.g. households tasks such as cooking) and societal knowledge (historical episodes, socio-historical knowledge, and contemporary events). Since all participants were in MUHAI, this ensured consistency and relevance to the use cases, yielding focused insights; however, this approach possibly limits the generalisability of the findings to other domains, which could be explored in future work. The open unstructured interviews took 20 minutes on average. The starting point of the interviews were the following questions:

1. What elements are needed to represent your (domain-specific) narratives? Why and what do you need to represent with these? [7]

2. What kind of data structure is required to represent your narratives?

[6] Meaning and Understanding in Human-Centric AI: `https://muhai.org`

[7] From a KR perspective, expressions that do not very easily fit into the regular <s,p,o> representation of RDF, like higher-order or n-ary relations.

3. For which tasks do you use your narratives? E.g. querying, visualisation.

4. Can you provide one example of a narrative in your use case?

Most participants had a computer science background. If not, we reformulated in non-technical terms. Table 2.1 provides a description of the interviews, with the domains, all related to MUHAI, the type of input data used to build the narrative, one example of a narrative and a practical application scenario for each narrative graph to be built. In the history and in the cognitive science domains (I1, I6), the main elements to be represented can be seen as extended version of the five Ws: Who, What, Where, When and Why. For the biomedical science (I2), the knowledge to be represented is much more structured and concise. The main difference for the social media analysis (I3) is to enable the representation of conflicting viewpoints, that is important when analysing discourse in social media. Arts & Literature (I5) additionally require context and background knowledge to enhance the comprehension as the text or the image is read. Lastly, the Robotic Kitchen (I4) involving a cooking robot in a virtual simulation needs to plan its actions to execute a recipe. In this context, each action is seen as an episode in a narrative. We detail hereafter the 6 main requirements we gathered and organized from these interviews.

Table 2.1: Interview Descriptions. #: number of people interviewed.

Id	#	Domain	Input Data	Narrative Example	Application scenario
I1	1	History	Historical events	Main **happenings** during the French Revolution [28]	**Query-answering** about historical events
I2	1	Biomedical Science	Scientific Claims	**PICOs** (Population, Intervention, Comparison and Outcomes) [313]	**Reasoning** and decision support in clinical research
I3	2	Social Media Analysis	User-generated content	**Claims** by users on Twitter regarding the Russo-Ukrainian conflict [307]	Analyzing and deriving **insights** from user-generated content and trends
I4	2	Robotic Kitchen	Recipes	**Robot execution** of a recipe [268]	Robotic **cooking task execution** based on recipe understanding
I5	2	Arts & Literature	Text (linguistics, semantics) & Images (paintings)	**Questions** arising when reading a text/painting, that form a narrative network together with the answers to these questions [311]	**Question-answering** to enable interpretive analysis from artistic or literary works
I6	1	Cognitive Science	Mental Models	Importance of **values** in our beliefs [341]	Exploring and **modeling** belief systems and cognitive processes

- **Agents**[8]**.** These are animate entities like humans or animals.
  - *Types of agents* distinguish between humans, animals, animate objects, etc. Occasionally, people made the distinction between an active agent, who acts with a goal in mind, and a passive agent, who is more a spectator in a scene.
  - *Properties* describe agent attributes, such as physical characteristics.
  - *Roles* describe roles of agents in the events.
- **Objects.** These are physical or abstract objects that do not fall under the category of agents.
  - *Types of objects* distinguish between different types of objects.
  - *Properties* describe objects attributes, such as physical characteristics.
  - *Roles* describe roles of objects in events.
- **Locations.** Places or settings in which events happens.
  - *Relations between locations* link locations between them.
- **Provenance.** This is a description of the origin of the data or statements.
  - *Narrative text* includes any original text narrative content.
- **Events and Actions.** Throughout the interviews, events appeared as the core element in the narrative. Similarly to agent, there is a distinction between an action and an event. An action happens because of the execution of a function by an agent, whereas an event simply happens. An action can be associated with pre-conditions and post-conditions, that enable the happening of the action, or result from the action.
  - *Types* distinguish between different types of events and actions.
  - *Properties* describe generic attributes, such as timestamps and locations.
  - *Relations* describe relations between events, including:
    - Mereological. Relations between parts and the wholes they form.

[8] In these requirements, we differentiate agents from objects, with agents not necessarily participating in events directly, but their participation would be modeled through relationships.

    - Temporal. Temporal relationships between events, such as the ones from Allen's Algebra [9].
    - Causal. Causal relationships between events.

- **Narrative Dynamics.** The higher-level elements that bind, influence, and drive the progression of the narrative. They play a critical role in shaping the overall story, providing the framework within which the narrative elements interact and evolve. They provide a type of narrative by bridging together elements from the above categories.
  - *Script* is a pattern of sequences of events that are related, like the restaurant scene [288]. This includes archetypes, plans, frames and processes.
  - *Storyline* is a coherent sequence of events, it is an instantiation of a script. This includes scenes, acts, episodes, sequences of events and situations.
  - *State* is the particular condition that someone or something is at a specific time or during a specific time interval.
  - *Goal* is a state that is desired by an agent.
  - *Outcome/Effect* is a state that results from a causal relation, representing the new condition after an event.
  - *Perspectives* are different interpretations of the same event.
  - *Conditions* include conditions under which events or actions can happens. This includes pre-conditions, results or post-conditions, complete-condition, coherence-condition, and enablement.

2.2.2 Narrative Requirements Representation

In Section 2.2.1, we identified requirements for including semantic concepts in the representations of narratives across a broad range of domains. In this section, we map these representations to a variety of syntactic constructs for KGs. We first describe each KR formalism, ordered by representation complexity. The complexity level is based on the introduction of new abstraction levels, the ability to handle more entities, or the incorporation of more sophisticated reasoning.

- **basic:** only binary relations without variables;
- **n-ary:** relations involving more than two entities;
- **variables:** placeholders that can be replaced with actual values or entities;

- **higher-order:** representations that involve functions or predicates applied to other functions or predicates;
- **temporal relation:** relationships that involve time aspects;
- **epistemic:** relates to knowledge and beliefs about knowledge;
- **causal:** cause-and-effect relationships between entities or events;
- **rules:** logic-based statements that define conditions and implications.

The left part of Table 2.2 summarises the mapping of narrative requirements to the interviews. The most frequent requirements are events and actions, agents, location, and time. The frequency of (physical) objects is low, except in specific use cases such as Robotic Kitchen and Arts & Literature. The right side of Table 2.2 maps narrative requirements to their most suited KG representation forms. Objects, locations, and agents require only basic or n-ary representations. Provenance requires higher-order representations, while most events are simple except for temporal and causal relationships. Narrative dynamics demand the most complex representations, involving variables, higher-order, epistemic, or rules. Objects, locations, and agents are therefore the easiest to represent, while events and narrative dynamics require more advanced KR formalisms.

Lastly, Table 2.3 provides a mapping of KR formalisms (cf. Table 2.2) to various syntactic constructs (KG, query, rules). The left part of Table 2.3 describes the different existing types of KGs. The right part of Table 2.3 shows which KR formalisms can be used to represent each construct. Since some KGs are additive, e.g. blank nodes contain simple KG triples, we focus on the additional complexity introduced by the more advanced KG constructs described in the "Description" column. The table highlights how different KR formalisms, ranging from basic structures like simple triples to more complex constructs such as rules and named graphs, require corresponding sophisticated KG representations.

2.3 Can existing ontologies represent narratives?

To answer **RQ2** on assessing whether ontologies can represent narrative requirements, we conduct a systematic review on event-centric and narrative-centric ontologies. We then evaluate the ontologies based on how effectively they can represent narrative requirements and based on their vocabulary usage.

Table 2.2: Mapping of narrative requirements to the interviews and their required KG representations. For a requirement r and interview I_i, a ✓ indicates that c was identified as a requirement in I_i. For a requirement r and a representation r, a ✓ indicates that c can be represented with r. Req: Requirements.

	Interviews							**Syntactic Constructs**								
Narrative Req.	**I1**	**I2**	**I3**	**I4**	**I5**	**I6**	**Total**	**basic**	**n-ary**	**variables**	**higher-order**	**temporal**	**epistemic**	**causal**	**rules**	**Total**
Agents	✓	✓	✓	✓	✓	✓	6	✓	✓							2
types	✓		✓				2	✓								1
properties	✓		✓				2	✓								1
roles	✓		✓	✓	✓		4	✓	✓							2
Objects				✓	✓		2	✓	✓							2
types				✓	✓		2	✓								1
properties				✓	✓		2	✓								1
roles				✓	✓		2	✓	✓							2
Locations	✓		✓	✓	✓		4	✓	✓							2
relations			✓	✓			2	✓	✓							2
Provenance	✓		✓		✓		3				✓					1
narrative text	✓		✓		✓		3				✓					1
Events/Actions	✓	✓	✓	✓	✓	✓	6	✓	✓			✓		✓		4
types	✓			✓			2	✓								1
properties	✓						1	✓								1
relations	✓	✓	✓	✓	✓	✓	6	✓	✓			✓		✓		4
Dynamics	✓	✓	✓	✓	✓	✓	6			✓	✓		✓		✓	4
script	✓		✓	✓	✓		4			✓					✓	2
storyline				✓			1				✓					2
state		✓		✓	✓		3				✓					1
goal			✓	✓			2				✓		✓			2
outcome/effect						✓	1				✓		✓			2
perspective			✓	✓	✓		3				✓		✓			2
conditions	✓		✓	✓			3			✓					✓	2
Total	**14**	**5**	**15**	**18**	**15**	**5**		**14**	**6**	**3**	**8**	**2**	**4**	**2**	**3**	

2.3.1 Systematic Review

We conducted a systematic review to compare existing ontologies based on their ability to represent narratives in a structured format. The process we used is illustrated in Figure 2.1. We used the query *"narrative OR event-centric"*, focusing on abstracts and titles. We then incorporated references from survey papers identified during the search or from other relevant sources, and

Table 2.3: Mapping of KG/Query/Rules representations to KR formalisms. NG: Named Graph, LPG: Labelled Property Graph.

Name	Construct	Repr.	Description	KR formalisms: basic	n-ary	variables	higher-order	temporal	epistemic	causal	rules
Simple KG	KG	RDF	simple (s, p, o) triples	✓				✓		✓	
Blank Node			nodes with no IRI		✓	✓					
Singleton Property			triples annotated		✓						
Reification			statement nodes		✓		✓	✓	✓	✓	
NG		NG	triples grouped				✓				
LPG		LPG	key-value pairs				✓	✓	✓		
Hyperrelational Graph		RDF-Star	«s p1 o1» p2 o2				✓	✓	✓		
Query	Query	-	-			✓					
Rule	Rule	-	-					✓	✓		✓

included ontologies cited in the identified papers. Survey papers included surveys on narrative-centric ontologies [339] and event-centric ontologies [267, 290, 294]. We included only papers that presented a novel ontology relevant for representing a narrative graph, with inclusion criteria determined through abstract review.

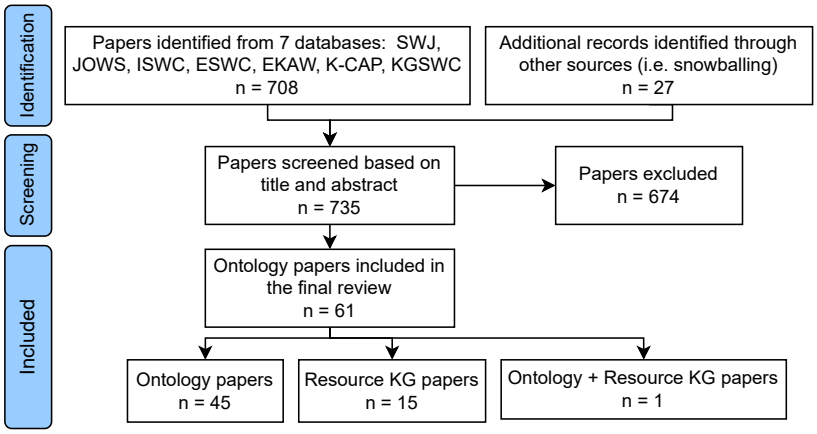

Figure 2.1: Process used to retrieve papers for our comparison (PRISMA [255]). For the snowballing, we looked at paper references from the first iteration.

This resulted in 61 papers: 45 were papers presenting an ontology, 1 presented both an ontology and a resource KG, and 15 were resources based on these ontologies. Among the 46 $(45 + 1)$ ontologies, we further analysed the

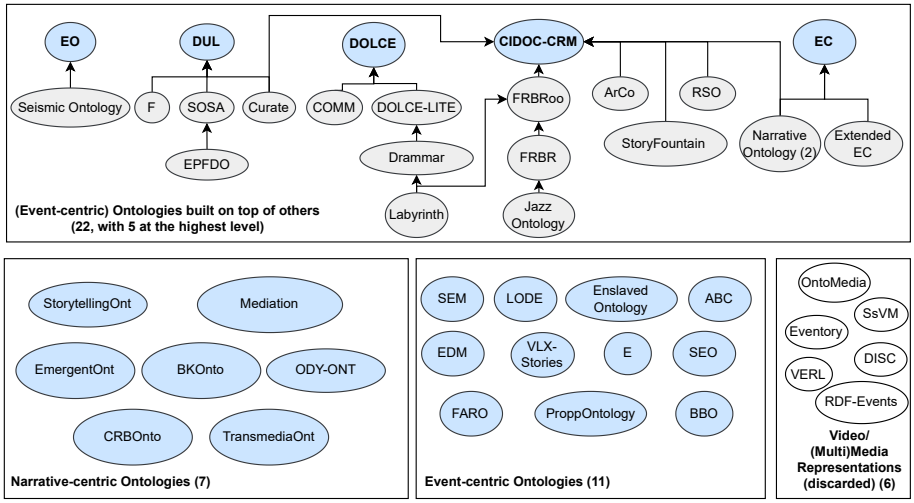

Figure 2.2: The connections between the various event-centric ontologies (1). The ones coloured in blue (from 1a) are the ones we kept for our comparison: EO [272], DUL [124], DOLCE [123], CIDOC CRM [94] and EC [192].

Table 2.4: Resource Paper Description.

Id	Resource	Ontology
1	Structured Event Entity Resolution [183]	EO [272]
2	VLX-Stories [108]	VLX-Stories [108]
3,4	EventKG [134, 135]	SEM [338]
5	DIVE [84]	SEM [338]
6	ECG [279]	SEM [338]
7	Wind in Our Sails [292]	CIDOC CRM [94]
8	BiographySampo [168]	CIDOC CRM [94]
9	ArCo [55]	CIDOC CRM [94]
10	SCUPLTEUR [1]	CIDOC CRM [94]
11	Using Event Spaces [244]	CIDOC CRM [94], LODE [294]
12	EventMedia [186]	CIDOC CRM [94], LODE [294]
13	OceanGraph [375]	SOSA [171] (extended from DUL [124])
14	3cixty [329]	DUL [124], Schema [289]
15	History of Vienna [193]	Schema [289]

other ontologies on which they were built, such as DUL [124] or CIDOC CRM [94]. This resulted in three categories of ontologies: (1) event-centric ontologies, which includes (1a) top-level ontologies and (1b) ontologies extended from ontologies from (1a),(2) narrative-centric ontologies, which in-

cludes (2a) narrative-centric ontologies and (2b) ontologies extended from ontologies from (2a), and (3) media-centric ontologies, which we discarded for our comparison since these were about video content that we did not have in our interviews. Table 2.4 and 2.5 describes the 15 resources and 46 ontologies respectively. Figure 2.2 shows the visual dependencies on the ontologies from category (1b) and (2b).

Table 2.5: Ontology Paper Description. *Extended from* specifies the original ontology, if applicable, from which the listed *Ontologies* were derived.

Type	Id	#	Ontologies	Extended from
Event	1a	16	EO [272], DUL [124], DOLCE [123], CIDOC CRM [94], EC [192], SEM [338], LODE [294], the Enslaved Ontology [296], ABC [198], EDM [96], VLX-Stories [108], E [356], SEO [107], FARO [275], ProppOntology [258], BBO [194]	-
Event	1b	1	Seismic Ontology [232]	EO [272]
		3	F [290], SOSA [171], EPFDO [234]	DUL [124]
		1	Curate [243]	DUL [124], CIDOC CRM [94]
		2	COMM [14], DOLCE-LITE	DOLCE [123]
		1	FRBR [251]	FRBRoo [95]
		1	the Jazz Ontology [270]	FRBR [251]
		1	Extended EC [59]	EC [192]
Narrative	2a	7	RST [246], Mediation [180], Princess Lovely [319], StoryTeller [372], ODYONT [184], DL Ontology for Fairy Tale [259], Transmedia Fictional Worlds [43]	-
Narrative	2b	1	Drammar [79]	DOLCE-LITE
		1	AO [78]	Drammar [79], FRBRoo [95]
		4	FRBRoo [95], ArCo [55], StoryFoutain [242], RSO [252]	CIDOC CRM [94]
		1	Narrative Ontology [22]	CIDOC CRM [94], EC [192]
Media	3	6	OntoMedia [200], SsVM [100], Eventory [349], DISC [129], VERL [119], RDFEvents [315]	-

We refer to our supplementary online material[9] for further details on the systematic review. The main document contains a description of our methodology and sources, and the folders contain the papers retrieved with the search engines. The ontologies highlighted in blue in Table 2.5 are the ones we retained for our comparison, i.e. the ontologies from categories (1a) and (2a). All the ontologies from (1b) and (2b) were extended from others from (1a) and (2a), so we focus on (1a) and (2a) ontologies instead. Table 2.6 describes the references found from the search engines: "all" refers to the number of results

[9] `https://bit.ly/ekaw-2024-systematic-review`

we got from the search queries, and "useful" refers to the number of references we kept.

Table 2.6: Provenance description of papers from the systematic review. N: Narrative, E: Event-centric. For *K-CAP/N (useful)*, 5 papers were not accessible.

Provenance	N (all)	N (useful)	E (all)	E (useful)
SWJ	7	2	2	2
JOWS	21	2	147	7
ISWC	35	9	66	2
ESWC	32	2	106	1
EKAW	16	1	14	1
K-CAP	44	1	176	2
KGSWC	2	1	40	4

2.3.2 Description of the Retained Ontologies

We included models from a survey on narrative-centric ontologies [339], that reviewed 11 ontologies that explicitly mention the term "narrative" in their description. [246] proposes a storytelling ontology using RST [228] that describes events and objects, and how they are connected. Both [22] and [242] are extended from CIDOC CRM [94], an ontology widely used in the cultural heritage domain. [22] focuses on representing core notions in narratology such as fabula, plot and narration, and [242] provides an interface to discover links between entities. A follow-up of [22] is the Narrative Ontology [235] to be used in Digital Libraries. The Mediation Ontology [180] is in the domain of international relations. The Labyrinth system [78] allows to discover narratives between entities in the domain of digital humanities and western archetypes. It re-uses ontologies like Drammar [221] and FRBRoo [95]. [319] presents a fabula model for emergent narratives, and re-uses the General Transition Network [328]. The main focus is to represent causal transitions between events. ODY-ONT [184] is used to query a literary text, and BKOnto [372] to represent biographical narratives. The CRBOnto ontology [259] relies upon Propp's morphology of folks stories [269]. [43] proposes an ontology to represent transmedia fiction such as the Marvel universe. [79] proposes a model based on Drammar [221] and DOLCE-LITE to represent actions motivated by heroes in the literature.

Event-centric ontologies encompass diverse domains such as: SEO [107] for scientific events, CIDOC CRM [94] and EDM [96] for cultural heritage, SEM [338], LODE [294], ABC [198], EO [272], EC [192] and E [356] for

more generic usages. CIDOC CRM [94] is widely used for the cultural heritage domain, and several other ontologies have since been derived from, such as FRBR [251] for bibliographic records. DUL [124] is a combination and a simplification of DOLCE [123] and the DnS [124] ontologies. Several ontologies were extended from DUL, such as F [290], originally for emergency response and SOMA-NARR [268]. The Enslaved Ontology [296] integrates content about the historical slave trade from diverse sources, while VLX-Stories [108] structures text from media feeds into an event-centric ontology. ProppOntology [258] is built upon Propp's morphology of the folktale [269]. Lastly, BBO [194] represents dynamic BPMN process executions in structured format and FARO [275] focuses on event relations.

2.3.3 Evaluation of Narrative Ontologies

Background for Ontology Evaluation.

Tim Berners-Lee defined a 5-star ranking schema to assess the quality of Linked Data [23]. This scoring system was based on four rules which did not include any considerations with respect to the vocabulary that includes broader categories such as ontologies and thesauri. [172] therefore proposed a new 5-star system to evaluate the quality of the vocabulary, displayed in Figure 2.3. We re-use this 5-star system to compare the ontologies.

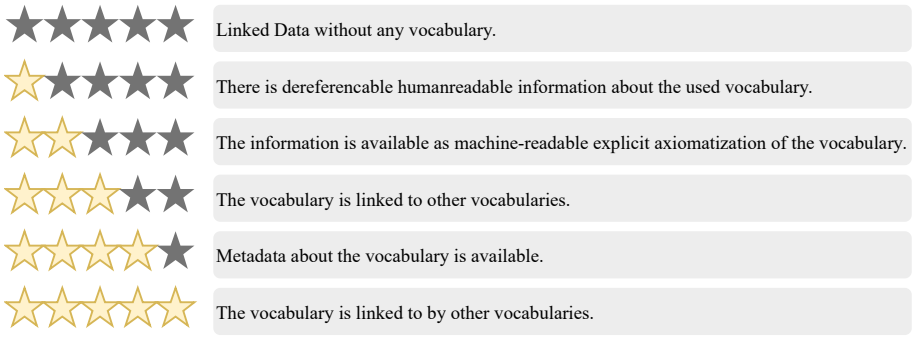

Figure 2.3: The 5-star system to evaluate ontologies from [172].

Analysis of the Ontologies.

Table 2.7 illustrates the coverage of our narrative requirements by the narrative and event-centric ontologies. The ontologies are listed in descending order of their 5-star scores [172], and subsequently in descending order of the number

Table 2.7: Coverage of narrative requirements for narrative and event-centric ontologies. A ✓indicates that requirement *r* can be represented in ontology *o*. Stars show the 5-star ranking from Section 2.3.3. E: event-centric, N: narrative-centric. Ontologies marked with * are unofficial designations used for clarity.

	★★★★★						★★★★☆							★★★☆☆			★★☆☆☆		★☆☆☆☆					☆☆☆☆☆	
Narrative Requirements	***DOLCE* [123]**	***DUL* [124]**	***CIDOC CRM* [94]**	***EDM* [96]**	***EO* [272]**	***EC* [192]**	***ProppOntology* [258]**	***Enslaved* [296]**	***SEO* [107]**	***SEM* [338]**	***Transmedia** [43]**	***FARO* [275]**	***LODE* [294]**	***ABC* [198]**	***ODY-ONT* [184]**	***Storytelling** [246]**	***BBO* [194]**	***CRBOnto* [259]**	***VLX-Stories* [108]**	***Labyrinth** [78]**	***E* [356]**	***Mediation* [180]**	***Emergent** [319]**	***BKOnto* [372]**	**Total**
Agents	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		21
types	✓	✓					✓	✓	✓				✓		✓		✓		✓	✓					10
properties	✓	✓						✓					✓		✓										5
roles	✓	✓					✓	✓		✓					✓	✓		✓							8
Objects	✓	✓	✓	✓			✓				✓			✓				✓	✓	✓					10
types	✓	✓	✓	✓			✓				✓			✓				✓	✓	✓					10
properties	✓	✓	✓	✓			✓				✓			✓				✓	✓	✓					10
roles	✓	✓					✓											✓	✓						5
Locations	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓		✓	20
relations			✓	✓	✓				✓												✓				5
Provenance	✓	✓	✓	✓				✓	✓								✓		✓		✓	✓			10
narrative text	✓	✓		✓				✓	✓										✓						6
Events/Actions	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	23
types	✓	✓	✓				✓	✓	✓	✓		✓		✓			✓		✓			✓			12
properties										✓															1
relations	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓		✓	✓	✓					✓	✓	✓	✓	17
Dynamics	✓	✓																							2
script	✓	✓																		✓				✓	4
storyline	✓	✓		✓			✓				✓					✓	✓								7
state	✓	✓	✓									✓								✓	✓				6
goal	✓	✓					✓					✓											✓		5
outcome/effect	✓	✓																					✓		3
perspective	✓	✓								✓													✓		4
conditions		✓	✓									✓											✓		4
Total	21	21	11	11	5	2	13	10	9	7	7	5	5	8	7	5	7	7	11	9	7	6	6	4	0
Event vs. Narrative-centric	E	E	E	E	E	E	E	E	E	E	N	E	E	E	N	N	E	N	E	N	E	N	N	N	
Domain-specific			✓	✓			✓	✓	✓		✓				✓	✓	✓	✓	✓	✓		✓	✓	✓	15

of requirements they cover. Each column in the table shows its 5-star score, how effectively an ontology fulfills the narrative requirements, whether the ontology is event-centric or narrative-centric, and whether it is domain-specific. We hereafter detail the main findings from Table 2.7.

Simple requirements with event-centric ontologies. For narratives with simple requirements such as event without intricate relationships or narrative dynamics, an event-centric ontology like SEM [338] may suffice. Event-centric ontologies generally offer higher quality in terms of 5-star score, but tend to represent less complex narrative requirements, including basic relationships between events and narrative dynamics.

The case of DOLCE [123] and DUL [124]. They can represent the broadest range of requirements, albeit at the cost of complexity and readability in terms of representation [275], since these are upper-level ontologies. By design, they are made to represent any domain, but users would need to define their own templates for specific applications or requirements, such as perspectives, which could be time-consuming.

Domain-specific ontologies. Apart from DUL [124] and DOLCE [123], the ontologies that can represent the most requirements tend to be domain-specific ontologies, in domains such as the digital humanities [94, 96] and literary texts [258]. This is good if one wants to build a narrative in this particular domain, but might be challenging to extend to other use cases.

Ontologies with lower 5-star rating. Ontologies such as Mediation [180] or Emergent*[10] [319] adhere less to established web standards, which not only lowers their 5-star scores but also significantly their interoperability and integration.

Narrative Dynamics. Fewer ontologies can represent the narrative dynamics, that are core to binding and driving the progression of the narrative elements. Among the 4-rated domain-specific ontologies, SEO [107] and Transmedia*[43] can represent scripts and storylines. Among the 4-rated generic ontologies, SEM [338] can represent perspectives through the `sem:View` class and blank nodes, and FARO [275] goals, states and conditions. The other ontologies that can represent narrative dynamics have lower 5-star ranking (cf. 1-score models in Table 2.7).

[10] Cf. Table 2.7 for the explanation on the asterisk.

KR Formalisms. In practice, referring back to Table 2.3, only basic RDF and some form of reification are employed in these ontologies. Reification allows for representing more complex relationships and n-ary structures, but can become verbose and increase modeling complexity, echoing similar issues noted for DOLCE [123] and DUL [124]. Together, these constructs can, in principle, support a variety of formal representations, excluding variables (primarily used in queries) and rules.

Other. The number of requirements an ontology can represent varies significantly, from 21 in DUL [124] and DOLCE [123] at the cost of complexity, to 2 in EC [192], although both with a high 5-star score. There is a positive correlation between key requirements, such as agent or event representation, and their presence in ontologies. It is important to note that the presence of a checkmark (✓) indicates the representation of a requirement but does not reflect the number of such relations an ontology can represent. For instance, some ontologies only include basic relationships between events, while others offer a more extensive list. Lastly, the simpler narrative requirements such as agents and objects are well covered by event-centric ontologies, whereas the more complex ones like narrative dynamics tend to be less covered or less adhering to web standards, highlighting the ongoing challenges of narrative representation in structured format.

2.4 Recommendations

To answer **RQ3** on the best ontology to represent a narrative, we provide generic recommendations on how to best identify a structured representation for a narrative. We lastly provide tailored recommendations from the interviews use case (cf. Tables 2.1 and 2.2).

2.4.1 Generic Recommendations

Based on the previous analyses on the narrative requirements and their KR formalisms (Section 2.2), and on how effectively ontologies fullfill these narrative requirements (Section 2.3), we recommend using the following steps to effectively build a narrative in a structured format for a specific application:

1. Refer to Table 2.2 to **identify the narrative requirements** that are relevant to your project. These tables categorise narrative requirements by their complexity, aiding in the selection process.

2. Based on the chosen narrative requirements, identify the **types of KR formalisms** needed from Table 2.2 and 2.3. This will help in understanding the level of complexity your model must support.

3. Choose **ontologies that can effectively represent** the identified narrative requirements. Use Table 2.7 to find ontologies that cover your narrative requirements and prioritise ontologies that have a high score on the 5-star schema ranking [172].

4. **Extend existing ontologies** by picking the one that fits best, and ensure it adheres to web standards to facilitate future integration, reuse and collaboration.

5. **Consider developing your own ontology** if existing ones are insufficient, and ensure it adheres to web standards to facilitate future integration, reuse and collaboration.

2.4.2 Tailored Recommendations from Interviews Use Cases

Table 2.8 provides an overview of the recommendations we give. **I1 and I3**, within the History and Social Media domains respectively, share numerous narrative requirements. The primary distinction is that I1 includes more generic event descriptions, while I3 encompasses more narrative dynamics such as goals and perspectives. Our recommendations are tailored to the type of KG employed, each with its own advantages and drawbacks. For example, reification is compliant with RDF standards but may be inefficient. If using reification exclusively, we suggest employing the core classes of DOLCE [123] or DUL [124], whose ontologies and design patterns are based on n-ary relationships. For I3 and the perspectives, SEM [338] is recommended due to its existing implementation. Other KGs, such as Named Graphs, are more compact but can become overloaded with numerous identifiers, while RDF-star is more efficient but still not widely adopted. In these cases, we recommend using DOLCE [123] or DUL [124] for simple narrative requirements and extending them to accommodate higher-order relationships. Additionally, we advise considering FARO [275], which, to our knowledge, can represent a wide range of event relations, including the goals pertinent to I3.

I2 and I6, within the Biomedical Science and Cognitive Science domains respectively, both have a limited number of narrative requirements. The primary difference is that the input data for I2 is already structured according to the PICO ontology[11], whereas the input data for I6 is unstructured. Given

[11] `https://linkeddata.cochrane.org/pico-ontology`

Table 2.8: Recommendations from interview use-cases.

Id	Domain	Recommendation
I1	History	DOLCE [123]/DUL [124] if reification, FARO [275]
I2	Biomedical Science	PICO with FARO [275]
I3	Social Media Analysis	Same as I1 + SEM [338] for perspectives
I4	Robotic Kitchen	DUL [124] or derived ontologies (as per the literature)
I5	Arts & Literature	ProppOntology [269], extending Emergent* [319] if time permits
I6	Cognitive Science	Extend or create an ontology

their limited conceptual needs, we recommend targeting ontologies that precisely represent these requirements rather than selecting the most comprehensive ones. For I2, which requires agents and relationships between events and states, we suggest combining PICO with FARO [275]. FARO is capable of representing numerous relationships between events and states, while PICO provides domain-specific vocabulary for agents. For I6, which requires agents, relationships between events, and outcomes/effects, Table 2.7 indicates that this last requirement is not well-represented across ontologies. The options available are the generic DOLCE [123]/DUL [124] or the domain-specific Emergent* [319]. Designing new patterns over DOLCE/DUL may be too complex for those not specialized in the semantic web. Therefore, we recommend either extending existing ontologies by creating new classes or properties, or building a new one tailored to these specific requirements.

I4 in the Robotic Kitchen domain has the most narrative requirements, including objects and multiple narrative dynamics, and is closely linked to robotic sensory data [25, 26]. Most work, such as SOSA [171], builds on DUL [124]. In line with the existing work, we recommend continuing using DUL [124] or derived ontologies.

I5 in the Arts and Literature domain overlaps with domains-specific ontologies in Table 2.7. We recommend starting with these ontologies. ProppOntology [269] covers most simple requirements and some complex ones like storyline. For other narrative dynamics such as outcome/effect or conditions, designing patterns over DOLCE [123]/DUL [124] is an option. However, given the domain-specific nature of this application, extending Emergent* [319] to cover more narrative dynamics and improve its sustainability is a preferable approach if time permits.

2.5 Discussion and Conclusion

In this paper, we map identified narrative requirements to appropriate knowledge representations. We conduct six interviews and identify core narrative requirements, as well as the KR formalisms they require. We find that requirements like objects or agents require simpler KR formalisms, while requirements such as events or narrative dynamics require more complex KR formalisms (RQ1). We only included domains relevant to MUHAI, limiting our findings to text-based or sensor-based narratives and potentially reducing applicability to other domains such as multimedia content, that could be explored in future work.

We then conduct a systematic review to extract event-centric and narrative-centric ontologies to represent narratives. We analyse the 24 retained ontologies with respect to how effectively they can represent narrative requirements and how well they are rated on the 5-star vocabulary system [172] (RQ2). We found that simple requirements are best represented by event-centric ontologies. Event-centric ontologies such as DUL [124] and DOLCE [123] cover a broad range of requirements but this comes at the cost of complexity [275], since these are upper-level ontologies, which means that one might need additional work to define templates. Lower-rated ontologies struggle with interoperability due to poor web standards adherence. Fewer ontologies effectively represent narrative dynamics, with SEM [338] and FARO [275] being notable exceptions with high ratings for goals, perspectives and conditions. Coverage of complex requirements is more limited and often poorly standardized.

We lastly provide recommendations on how to best pick a model to represent a narrative, and apply them to the interview s (RQ3). The interview subjects and domain-specific ontologies cover key domains of narrative representation, but the list is not exhaustive, potentially limiting generalizability. Our semantic-web focus may narrow the scope of our recommendations to semantic web-related techniques, but broader exploration could lead to more comprehensive guidance across various KR paradigms.

In future work, we aim to enhance the evaluation of models by integrating other metrics for ontology evaluation, and to improve the generalisability of this study. Various works [44, 49, 97, 132, 159, 271, 343] have attempted to define criteria to evaluate an ontology from various perspectives such as Software Engineering, verification, validation or semiotics. Most importantly, we are actively working on automating the generation of structured narratives from KGs. Our initial efforts included a prototype for generating a narrative on the French Revolution [29] using the SEM ontology [338], where the data collection was performed manually.

3

Event-centric KG Construction

Based on [35]:
Inès Blin, Ilaria Tiddi, Remi van Trijp and Annette ten Teije
ChronoGrapher: Event-centric Knowledge Graph Construction via Informed Graph Traversal
Semantic Web Journal

The previous chapter investigated narrative requirements and their representation in knowledge models, evaluated ontologies for coverage and suitability, and derived practical recommendations. We now turn to use cases that build on these insights. In this chapter, we introduce the historical use case. We present a system for automatically constructing event-centric knowledge graphs. It combines semantically guided graph traversal with enrichment techniques.

Abstract. Event-centric knowledge graphs help enhance coherence to otherwise fragmented and overwhelming data by establishing causal and temporal connections using relevant data. We address the challenge of automatically constructing event-centric knowledge graphs from generic ones. We present ChronoGrapher, a two-step system to build an event-centric knowledge graph from grand events such as the French Revolution. First, a pruned, semantically informed best-first search traversal retrieves a subgraph from large, open-domain knowledge graphs. We define event-centric filters to prune the search space and a heuristic ranking to prioritise nodes like events. Second, we combine a structured triple enrichment method with a text-based triple enrichment method to build event-centric knowledge graphs. ChronoGrapher demonstrates adaptability across datasets like DBpedia and Wikidata, outperforming approaches from the literature. Furthermore, it is designed to be flexible and to operate over any knowledge graph accessible through HDT dumps

or SPARQL endpoints. To evaluate the utility of these constructed graphs, we conduct a preliminary user study comparing different prompting techniques for event-centric question-answering. Our results demonstrate that prompts enriched with event-centric knowledge graph triples yield more factual answers, measured by how well answers are grounded in source information, than those enriched with generic triples or base prompts, while preserving succinctness and relevance.

3.1 Introduction

We tackle the problem of building event-centric knowledge graphs (KGs) automatically starting from generic KGs. An event-centric KG can be seen as a KG that focuses on events as central entities, rather than generic entities. An event is an occurrence or happening that can be described by its attributes and relationships to other entities within a KG. It includes information such as time, location, participants, and other relevant properties. Relating events together provide coherence to otherwise fragmented and overwhelming data by establishing causal and temporal connections using relevant data. There are numerous scenarios in which systems could assist individuals in constructing event-centric KGs in their tasks: in news article retrieval for event-centric exploration [342], in the legal domain where experts must associate events that could bolster their case [110], or in the historical domain when generating hypotheses by retrieving valuable information from archives [238].

Let us imagine a historian seeking to understand the events that happened at the end of the French Revolution. Our system aims to produce an event-centric KG, as illustrated in Figure 3.1, which we manually created using mainly the SEM [338] ontology and Wikidata. Figure 3.1 shows that the *Coup of 18 Brumaire directlyPrecedes* the *French Consulate*, and that Napoleon was a *participant* in the *Coup of 18 Brumaire*, and *First Consul* during the *French Consulate*. We can infer that the *Coup of 18 Brumaire* marked the beginning of the *French Consulate* and enabled *Napoleon* to become *First Consul*.

It should be noted that the classification of entities such as the *French Directory* and the *French Consulate* can be ambiguous, as they may be interpreted either as governing bodies (i.e., institutional entities) or as historical periods marking distinct phases in political history. In our work, we choose to treat such entities as events, consistent with how they are often described in historical discourse (e.g., "the period of the French Directory") and represented in sources like DBpedia and Wikidata, where these entities frequently include temporal attributes (start and end dates) and are part of event-type hierarchies.

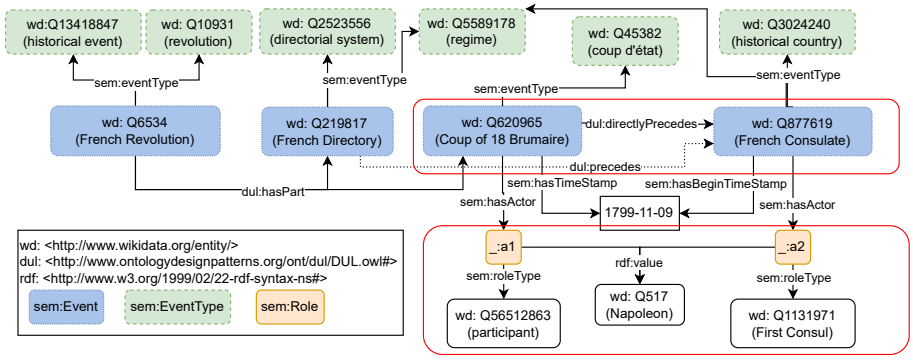

Figure 3.1: One example of an event-centric KG on the Coup of 18 Brumaire.

Previous work has approached event-centric KG construction as a two-step process: first, identifying relevant cues within a large memory or knowledge base (step-1), and second, analyzing and reasoning about these cues (step-2) [300]. Recent studies have also used generic KGs as background knowledge to support event-centric KG construction [139]. In this context, we approach event-centric KG construction using the following two steps: given an input event node in a KG, we extract an event-centric subgraph from that KG (step-1), and we enrich and link this subgraph with additional information to build a new event-centric KG (step-2). Each triple in the event-centric subgraph from step-1 has a sub-event of the input event as its subject or its object, making the extracted content directly relevant to the construction of the new event-centric KG in step-2. Most existing event-centric KG resources are however built from annotated text [204, 247, 279] or from generic KGs [135] (step-2), but do *not* focus on identifying related events for event-centric KG construction (step-1). Likewise, event subgraph extraction (step-1) has been used for downstream tasks such as question-answering (QA) [174, 209, 303], but does *not* focus on building event-centric KGs (step-2). We propose ChronoGrapher that both retrieves event-centric subgraphs and generates new event-centric KGs, as well as an evaluation of an event-centric KG constructed by ChronoGrapher in a QA setting in the form of a user study. ChronoGrapher, along with the source code and detailed instructions to reproduce the experiments, is publicly available on GitHub under the GPL-v3 license[1], and archived on Zenodo[2].

Our objective is to construct event-centric KGs that support downstream tasks such as QA, for example by enabling the summarization of complex events like the French Revolution or verifying whether figures like Napoleon and Paul Barras were both involved in the Coup of 18 Brumaire. We define three research questions:

[1] `https://github.com/SonyCSLParis/graph_search_framework/tree/main`
[2] `https://doi.org/10.5281/zenodo.7194677`

RQ1: How to extract an event-centric subgraph from a KG, given an input event node? (step-1)

RQ2: How to create an event-centric KG from that subgraph? (step-2)

RQ3: Can integrating event-centric KGs into a QA pipeline improve the output quality of large language models (LLMs) compared to using LLMs alone?

Figure 3.2 provides an overview of the different tasks we tackle, as well as the input and output of each component.

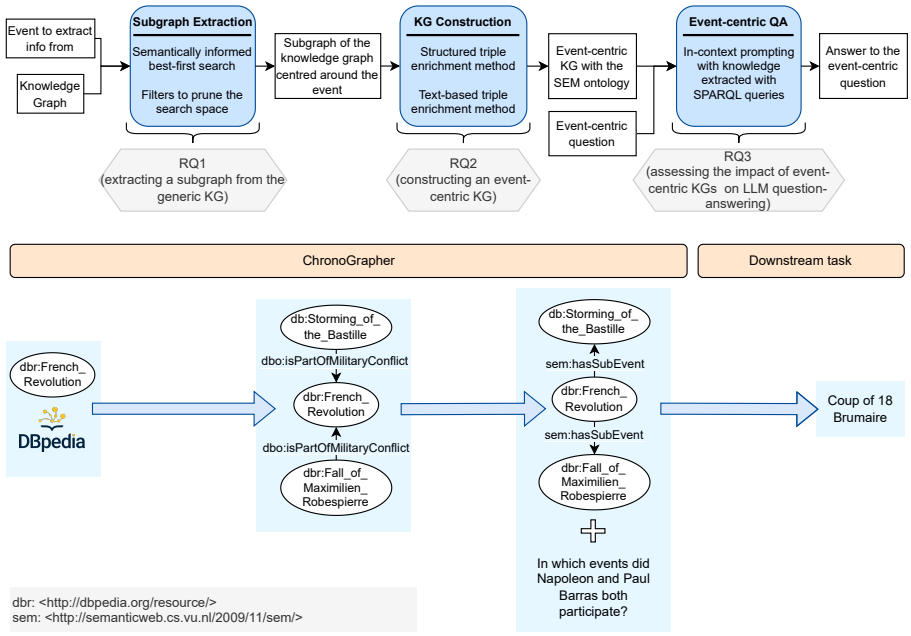

Figure 3.2: Overview of the diverse tasks we tackle to answer our research questions. The example on the bottom is with the French Revolution as starting node and DBpedia as the KG. ChronoGrapher integrates both a subgraph extraction step through a pruned, semantically informed best-first search traversal, and a KG construction step through a structured triple enrichment method and a text-based triple enrichment method.

For each of the research questions, we hypothesize the followings:

H1: Due to the vastness of KGs, exploring the entire graph is impractical. A search strategy can help efficiently extract the most relevant parts of the graph.

H2: Combining both coarse-grained (e.g., entity-level) and finer-grained (e.g. roles) information leads to a richer and more useful meaningful representation.

H3: Since KGs can be difficult to navigate or interpret directly, they are often not exposed to end users. Instead, their utility can be evaluated through downstream tasks such as QA, where improved performance indicates higher-quality structure and content.

To address these questions, we present ChronoGrapher, a framework composed of two components. First, a pruned, semantically-informed best-first search traversal mechanism extracts event-centric subgraphs around a given event node (RQ1). Second, a KG construction module converts this subgraph into an event-centric KG using information from both the structure and literals (RQ2). We then evaluate the resulting KGs through a QA task to assess whether integrating event-centric KGs enhances the performance of LLMs compared to using LLMs alone (RQ3).

For **RQ1**, ChronoGrapher performs an informed graph traversal to extract subgraphs composed of sub-events that are related to a given event node in the KG. It is a pruned, *semantically-informed* best-first search. First, it uses *event-centric filters* to prune the search space and to guide the exploration of the graph, and it identifies which ones are the most relevant to build an event-centric KG. Second, it adds a ranking stage to prioritise nodes for the best-first search part, which, to the best of our knowledge, has not been used in search systems starting from one node in KGs. We define the ranking heuristics based upon syntactic and semantic properties of the graph.

For **RQ2**, ChronoGrapher combines a structured triple enrichment method (or IRI-based method) with a text-based triple enrichment method (or abstract-based method) to generate an event-centric KG. The structured triple enrichment method is a rule-based system that converts the triples from the original KG into event-centric ones. The text-based triple enrichment method is linked to the valuable information embedded in textual literals, such as `dbo:abstract`. For this, we extract frame-based semantics [109] from these texts using semantic parsing. Frames can be seen as templates with placeholders to be filled in. For example, from the sentence *"The coup of 18 Brumaire ended the French Revolution"*, one could extract a `Causation` frame where *"coup of 18 Brumaire"* is the `Cause` and *"end of the French Revolution"* is the `Effect`.

Concerning the technical novelty of the work, most methods for representing events in KGs are underdeveloped, manually built and time-consuming, and ChronoGrapher is a method for building event-centric KGs automatically from a generic KG. Unlike most approaches, ChronoGrapher integrates data from existing KGs and texts, and results in more concise and precise outputs than baselines adapted from existing systems. The modularity of our approach ensures flexibility and adaptability, while the novel combination and adapta-

tion of previously unintegrated elements contributes to the field and improves the adapted baselines [117, 170]. We evaluate ChronoGrapher and adapted methods from the literature [117, 170] against ground truth events from EventKG [135]. ChronoGrapher shows adaptability and efficiency across datasets like DBpedia and Wikidata, outperforming existing methods.

For **RQ3**, we assess whether integrating event-centric KGs enhances the performance of LLMs compared to using LLMs alone. A human evaluation is conducted to compare three types of zero-shot QA prompts (without explicit context, with triples from generic KGs, with triples from event-centric KGs) and six types of different questions. Our results demonstrate that prompts enriched with event-centric KG triples yield more factual answers compared to those enriched with generic KG triples or base prompts, measured by how well answers are grounded in source information, achieving groundedness scores of 2.85, 2.24, and 1.11 respectively, while preserving succinctness and relevance.

3.2 Related Work

In this section, we review related work along three dimensions: (i) event-centric KGs and their usage, (ii) methods for event-centric KG construction from KG, and (iii) retrieval augmented generation. Recent advances have introduced new approaches to build event-centric KGs with various applications (i). In particular, we propose a method for event-centric KG construction, which directly contributes to this field. Unlike most previous methods that focus on constructing event-centric KGs from unstructured data, our approach builds KGs from existing, structured KGs. To achieve this, our aim is to improve how event-centric subgraph information is searched and navigated within input KGs, by optimizing existing graph traversal methods. To this end, we present related methods that facilitate this process (ii), including link-traversal-based techniques for dynamic data discovery, graph navigational languages for expressing complex patterns, and entity relatedness measures to prioritize relevant nodes. Lastly, we review works that utilize event-centric KGs to improve the performance of LLMs, highlighting the practical applications of event-centric KGs in downstream tasks (iii).

(i) Event-centric KGs and their usage. Recent research on KGs has emphasized the importance of having more event-centric structures to facilitate the analysis and understanding of sequence of events [277]. In this context, event-centric KGs have been developed in a wide range of domains, including news [204, 247, 279], political events [40], literary narratives [182], the biomedical domain [215], tourism [361], the legal domain [110], olfactory

heritage [212] and plenary debates [299]. In the digital humanities and cultural heritage domains, KGs are increasingly used to link objects to events and the stories surrounding them. For example, the Arco project constructs an Italian cultural heritage KG [55] that was built on XML data using the eXtreme Design methodology [36]. Other projects focus on understanding historical processes, whether through the history of a city such as Amsterdam [249] or through various historical archives [239, 298, 326, 336]. [336] focused on the extraction and modeling of events from the text archive, and [326] proposes an agile cataloging process to build a KG. These works primarily extract events from textual sources, often employing domain-specific heuristics or manual structuring.

In this work, we focus on KG construction from KG. The work most closely related to ours is EventKG [134], an event-centric KG that was built starting from large, generic KGs, such as DBpedia or Wikidata. EventKG follows a global and schema-driven approach, where event types are predefined and extracted in bulk across the entire KG. In contrast, our methods adopts a local and dynamic approach: we start from a given input event, that is a node in the KG, and automatically retrieves a subgraph that contains sub-events of that input event. This allows for more context-sensitive and targeted event graph construction. Moreover, our framework is designed to be more flexible: it can be applied to any KG, not limited to the largest open-domain KGs, and it can be configured to retrieve subgraphs around any type of node, not just events, making it broadly adaptable beyond the event-centric setting.

In addition to the KG construction part, event-centric KGs have been explored for downstream applications. For example, [143] improved sequential pattern mining with time and reasoning to analyze complex dynamical systems. Understanding such systems helps in acquiring new knowledge but also in predicting and controlling its behavior. Event-centric KGs also help in decision-making, for instance, in the epidemiological domain [19] or in maritime transport [87]. [195] lastly uses structured narrative representations to build hypotheses on knowledge bases. These applications underscore the growing importance of event-centric KGs in both analytical and predictive tasks.

(ii) Methods for Event-centric KG construction from KG. To construct event-centric KGs from KGs, our method performs a pruned, semantically informed, best-first search graph traversal. Although not originally designed for event-centric KG construction, several sub-domains offer valuable methods and insights that support this task. We draw on techniques from linked data traversal, graph navigational languages, and entity relatedness and summa-

rization to inform our approach. These methods guide how information is explored, filtered, and extracted from large graphs, which is crucial for retrieving event-centric content efficiently. We compare their scope to ChronoGrapher's.

Linked-traversal based methods focus on identifying and extracting task-relevant information by navigating large-scale web data. Also framed as Link Traversal Query Processing (LTQP), the engines start with seed URIs and dynamically discovers sources by following hyperlinks [147, 148]. One of the pioneering systems in this domain is LDSpider [170], which employs diverse search strategies such as breadth-first search or load balancing to crawl web data. Subsequent systems aimed to reduce the search space during the traversal, encoding features like property selection [287] or basic graph patterns [334]. Systems such as Linked Data-fu [309] and LiDAQ [334] and its extensions [335] added reasoning capabilities to their approaches. Other systems like SQUIN [146] focused on optimizing query execution efficiency. [225] proposes a hybrid SPARQL query execution strategy that combines link traversal with distributed query processing over SPARQL endpoints. A follow-up of their work [149] evaluated 14 ranking approaches to prioritize result-relevant data sources early during query execution. [321] extends LTQP by leveraging structural properties of decentralized systems such as Solid to improve query performance at scale. As a follow-up, [101] investigates the behavior of the link queue in LTQP and reveals two distinct execution patterns. Their findings highlight the importance of understanding link queue behavior for optimizing LTQP performance in decentralized settings. Lastly, Comunica [320] is a modular query engine for Linked Data, designed to flexibly support and evaluate diverse query algorithms, SPARQL features, and Web interfaces. These methods offer a variety of traversal strategies designed for efficient and goal-directed search through KGs, which we draw on to dynamically guide the extraction of event-centric subgraphs in our method.

Graph navigational languages allow structured and iterative exploration of Linked Data through declarative instructions to retrieve nodes or subgraphs. There are two types of approach, namely SPARQL extensions [116, 144, 276] and standalone languages [117, 150]. Most of these approaches use regular expressions, such as SPARQLeR [190] and PSPARQL [8], and later nested regular expressions, such as [376] and nSPARQL [263] to explore the graph. Although our traversal is not expressed in a declarative language, we compare the scope and output of our method to those of navigational languages. Our approach can handle both local KGs and SPARQL endpoints, and its output is a subgraph, referred to as a "region" in the context of GeL [115] and NautiLOD [117]. The most expressive system appears to be LDQL [150] which separates pattern processing and source selection. These approaches provide

techniques to extract nodes or subgraphs that match specific semantic criteria, which we draw on to refine and optimize the extraction of event-centric subgraphs in our framework.

Entity Relatedness methods support the identification of paths, "regions" or subgraphs of high semantic relevance within KGs [175]. Entity relatedness systems compute all paths with search heuristics and activation criteria to prune the search space and rank paths [158]. The search heuristics include the bidirectional search [105, 325], backward search heuristic [157], SPARQL queries [266], exploratory association search [70], and A* [86]. Activation criteria include entity clustering [70], metrics based on node degree [105, 241] or ontology [220], and similarity measures [86, 157] such as Jaccard, Wikipedia link-based measures [360] or SimRank [173]. For ranking paths, metrics include PF-ITF [266], exclusivity-based relatedness [157] and point mutual information [167]. Table 3.1 provides an overview of the entity relatedness systems we described. **Entity Summarization** systems [58, 213] take a KG as input and output a graph that retains certain original properties of the graph. The methods used to select nodes are similar to the ranking ones from entity relatedness systems. These systems offer a range of search heuristics and path-ranking strategies to identify meaningful connections between nodes, which we adapt to dynamically guide the extraction of subgraphs in our framework.

Table 3.1: Overview of entity relatedness systems. "-" indicates that the component is absent in the system.

System	Path Search	Activation Criteria	Ranking
[325]	Bidirectional search	Primitives	Cost functions
Profiler [157]	Backward search heuristic	Similarity measures	-
RECAP [266]	SPARQL queries	-	Path informativeness
EXPLASS [70]	Exploratory association search	Clustering entities	-
[86]	A*	Jaccard-based metrics	Top k paths
[241]	Shortest paths	Low-degree nodes	Top k paths
REX [105]	Bidirectional search	Degree of node	Interestingness measure
RelFinder [220]	All paths	Class, link and length filter	-

Our method traverses KGs to extract event-centric subgraphs for event-centric KG building, similar to link traversal-based methods and graph navigational languages. It operates on local graphs or any SPARQL endpoint and produces meaningful subgraphs, similarly to systems like [115] and [117]. Unlike existing traversal methods, which often lack finer-grained control and expressiveness, our approach introduces flexible navigation and reasoning capabilities. It employs filters for search space pruning without prior knowledge of

predicates, unlike graph navigational languages requiring regular expressions. Additionally, we introduce a ranking stage inspired by entity relatedness and summarization systems, prioritizing nodes based on a cost function. To the best of our knowledge, this dynamic prioritization is only present in [149] in existing navigational systems, and only propose two graph-related approaches. In contrast to entity relatedness systems that typically compute all paths between entities, our system dynamically computes paths. Furthermore, our approach requires only one starting node as input, with target nodes defined as variables. It extracts subgraphs from the original KG, aiming to retrieve all events of a given entity, disregarding specific property considerations as in entity summarization systems. Lastly, while entity summarization techniques focus on preserving specific property distributions, our method targets the extraction of complete event-centric subgraphs, ensuring the retrieval of all sub-events linked to the input node.

(iii) Retrieval Augmented Generation (RAG). Studies have shown that LLMs store factual knowledge in their parameters [265], e.g. in the form of implicit knowledge bases [274, 278]. However, the mechanisms to retrieve and understand this knowledge remain challenging. RAG systems [65] explicitly provide external information to the model through a dedicated *knowledge retriever component*. By separating knowledge storage from generation, RAG systems improve interpretability and often lead to more factual, diverse, and contextually grounded output [206], with applications in tasks such as QA [63] or reasoning [317]. Systems such as ORQA [203] or REALM [142] learn the retriever and the reader jointly, whereas systems like DrQA [63] have a fixed document retrieval component. [179] proposes a dense passage retriever to index all documents in a latent space. [264] augments the reader with a retriever in an unsupervised way.

In our user study, we adopt a RAG-like setup to compare three types of zero-shot prompting strategies for QA: one prompting without explicit context, and two prompting with triples from generic KGs and event-centric KGs, respectively. Our *knowledge retriever component* is implemented using `SPARQL` queries that extract relevant triples from the KG to condition the LLM's generation.

3.3 ChronoGrapher

As displayed in Figure 3.2, ChronoGrapher takes an event IRI as input, extracts an event-centric subgraph where all triples contains sub-events of that input IRI, and outputs an event-centric KG. In this section, we formalize the two

main components: subgraph extraction (Section 3.3.1) and event-centric KG construction (Section 3.3.2). We begin by defining the core concepts used throughout the two components: knowledge graphs, event-centric knowledge graphs, event and subevent.

Definition 3.1 (Knowledge Graph)**.** A **knowledge graph** (*KG*) is an RDF graph where the nodes represent entities (subjects or objects) and literals (objects) and the edges represent the relationships (predicates) between these entities and literals. Formally, $KG = \{(s,p,o)|s \in E, o \in E \cup L, p \in R\}$, where E is the set of entities (nodes) in the KG, L is the set of literals in the KG, and R is the set of predicates (edges) representing relationships in the KG.

Definition 3.2 (Event-centric KG)**.** In this work, an **event-centric knowledge graph** is a KG whose vocabulary, classes and relationships originate from the SEM [338] and the NIF [153] ontologies. There are four core classes in SEM: `sem:Event` (what), `sem:Actor` (who), `sem:Place` (where) and `sem:Time` (when). The NIF ontology enables to integrate text data in KGs.

Definition 3.3 (Event and subevent)**.** An **event** is defined as an occurrence in time that can be characterised by attributes from the SEM ontology, such as actor and location. A **sub-event** is defined as an event that forms part of a larger, composite event. Sub-events maintain the same formal structure as events but are contextually linked to parent events through temporal, causal, or structural relations (e.g. `sem:subEventOf`).

3.3.1 Event-centric Subgraph Extraction (RQ1)

Given a target event of interest and a knowledge graph *KG*, our goal is to extract an event-centric subgraph from *KG* that captures relevant information about this event. To achieve this, we propose a link traversal-based method tailored to the event-centric setting. Our approach takes the form of a pruned best-first search: it incrementally explores the most promising nodes while explicitly pruning others, enabling focused and efficient traversal.

We begin by introducing the core concepts required for graph traversal (*Preliminaries*) and formally define the task and the search algorithm (*Task and Method Overview*). We then present the two key components of our method: the event-centric filters, which prune the search space by discarding nodes (*Event-centric filters*); and the ranking and scoring mechanism, which selects the highest scoring nodes for expansion based on their contextual relevance (*Ranking Mechanism*). Together, these components drive the extraction of compact and semantically meaningful subgraphs centered around a given

event. To facilitate understanding of the subgraph extraction task and its components, we summarize the key definitions used throughout this section in Table 3.2.

Table 3.2: Useful definitions for the subgraph extraction task.

Def. #	Def. Name	Relevance to Subgraph Extraction Task
4	Ingoing Triples	Node Expansion (cf. Def. 6)
5	Outgoing Triples	Node Expansion (cf. Def. 6)
6	Node Expansion	KG Traversal
7	Event-Centric Subgraph Retrieval Task	Task
8	Pruned Best-First Search	Method, with two main components: node pruning and node ranking
9	Event-Centric Filters	Node pruning (cf. Def. 8)
10	Relation Patterns	Node ranking (cf. Def. 8)
11	Scoring	Node ranking (cf. Def. 8)
12	Preference Function	Node ranking (cf. Def. 8)

Preliminaries

We formally define preliminary concepts that are useful for our subgraph extraction task and our link traversal-based method: the ingoing and outgoing triples of a node in a KG, as well as node expansion.

Definition 3.4 (Ingoing triples)**.** Let $KG = \{(s,p,o)|s \in E, o \in E \cup L, p \in R\}$, and $P(KG) = \{A|A \subseteq KG\}$ the set of all subsets of KG. The **ingoing triples** of a node n are formally defined as:

$$ingoing_{KG}\colon E \longrightarrow P(KG)$$
$$n \longmapsto \{(s,p,n) \mid \exists p \in R,\ \exists s \in E,\ \text{ such that } (s,p,n) \in KG\}$$

Definition 3.5 (Outgoing triples)**.** Let $KG = \{(s,p,o)|s \in E, o \in E \cup L, p \in R\}$, and $P(KG) = \{A|A \subseteq KG\}$ the set of all subsets of KG. The **outgoing triples** of a node n are formally defined as:

$$outgoing_{KG}\colon E \longrightarrow P(KG)$$
$$n \longmapsto \{(n,p,o) \mid \exists p \in R,\ \exists o \in E \cup L,\ \text{ such that } (n,p,o) \in KG\}$$

Definition 3.6 (Node Expansion)**. Expanding a node** n means retrieving its ingoing and outgoing triples.

Building on these preliminary concepts, we now formalize the event-centric subgraph retrieval task and describe our pruned best-first search algorithm to tackle it.

Task and Method Overview

We formalize the event-centric subgraph retrieval task and present the framework used to address it. Our method is a pruned best-first search that iteratively expands a local subgraph by exploring the most promising candidate nodes while discarding other ones. This section first introduces the task formulation and algorithm definition, followed by a step-by-step breakdown in pseudo-code and an illustration of the traversal process.

Definition 3.7 (Event-centric Subgraph Retrieval Task)**.** Given a knowledge graph $KG = \{(s,p,o) \mid s \in E, o \in E \cup L, p \in R\}$ and a starting event node n_{start}, the goal of the **event-centric subgraph retrieval task** is to extract a subgraph $G' \subseteq KG$ such that each triple $(s',p',o') \in G'$ satisfies the condition that either s' or o' is a sub-event of n_{start}[3]. Formally:

$$r_{KG} \colon E \to P(KG)$$
$$n_{start} \mapsto \left\{(s',p',o') \in KG \middle| \text{subevent\_of}(s', n_{\text{start}}) \vee \text{ subevent\_of}(o', n_{\text{start}})\right\}$$

Note that with this definition, n_{start} will not be included in the extracted subgraph if it is not explicitly encoded as a sub-event of itself. However, n_{start} is later re-integrated during the KG construction step (described in Section 3.3.2), and is therefore present as an event node in the final output KG. Furthermore, non-subevent relations such as preceding events are not included in the extracted event-centric subgraph from Definition 3.7, and are not treated as sub-events. Extending this approach to incorporate other relations, such as those defined in Allen's interval algebra [9], is a promising direction for future work.

Definition 3.8 (Pruned Best-first Search)**.** A **pruned best-first search algorithm** explores a graph by expanding the unvisited node with the highest score according to a predefined heuristic. During traversal, the algorithm prunes (i.e., excludes) nodes and edges of specific types, reducing the search space while focusing the exploration on the most promising parts of the graph.

[3] Given that s' would be a sub-event, we assume it is also an event (analogously for o').

Algorithm 1 Pruned Best-First Search Algorithm

Input:

- $KG = \{(s,p,o) \mid s \in E, o \in E \cup L, p \in R\}$: knowledge graph to traverse;
- n_{start}: starting node for the search;
- t_s, t_e: start and end dates of n_{start};
- $iteration$: number of iterations;
- $extract\_type$: node types to extract;
- $filtering$, $scoring$, $preference$: traversal components (defined in Section 3.3.1).

Output:

$$G' = \left\{ (s',p',o') \in KG \;\middle|\; \begin{array}{c} \big(\text{type}(s') \in extract\_type \wedge \text{subevent\_of}(s', n_{\text{start}})\big) \\ \vee \big(\text{type}(o') \in extract\_type \wedge \text{subevent\_of}(o', n_{\text{start}})\big) \end{array} \right\}$$

$FILTER \leftarrow \text{INITFILTER}(filtering, t_{\text{start}}, t_{\text{end}})$ # Filtering
$SELECTOR \leftarrow \text{INITSELECTOR}(scoring, preference)$ # Node Selector
$P \leftarrow \emptyset$ # Pending nodes, that could be expanded
$nodes \leftarrow \{n_{start}\}$ # Nodes to expand
$N_{expanded} \leftarrow \emptyset$ # Nodes expanded
$G' \leftarrow \emptyset$ # Output
for $i = 1$ to $iteration$ **do**
  for all $node \in nodes$ **do**
    $N_{expanded} \leftarrow N_{expanded} \cup \{node\}$
    $ingoing \leftarrow \text{INGOING}(node, KG)$ # Stage 1: Expansion
    $outgoing \leftarrow \text{OUTGOING}(node, KG)$
    $(to\_add, pending) \leftarrow FILTER(ingoing, outgoing, extract\_type)$ # Stage 2: Filtering
    $G' \leftarrow G' \cup to\_add$
    $P \leftarrow P \cup pending$ # Stage 3: Output update
  end for
  $nodes \leftarrow SELECTOR(P)$ # Stage 4: Ranking
end for
return G'

We provide the pseudo-code for our subgraph extraction method in Algorithm 1, which implements a pruned best-first search in four main stages at each iteration of the search: nodes with the highest score are selected and expanded in the graph (Stage 1); event-centric filters prune certain nodes from the search space (Stage 2); the pending set and output subgraph are updated accordingly (Stage 3); and nodes in the pending set are scored and ranked to guide the next iteration (Stage 4). The *to_extract* parameter specifies the type of nodes to extract from the graph; while it is set to events in our experiments, it can be adapted to extract different node types depending on the dataset or use case. The set nodes, initialized in line 4, represents the priority queue of nodes to be expanded, drawn from the pending set P initialized in line 3.

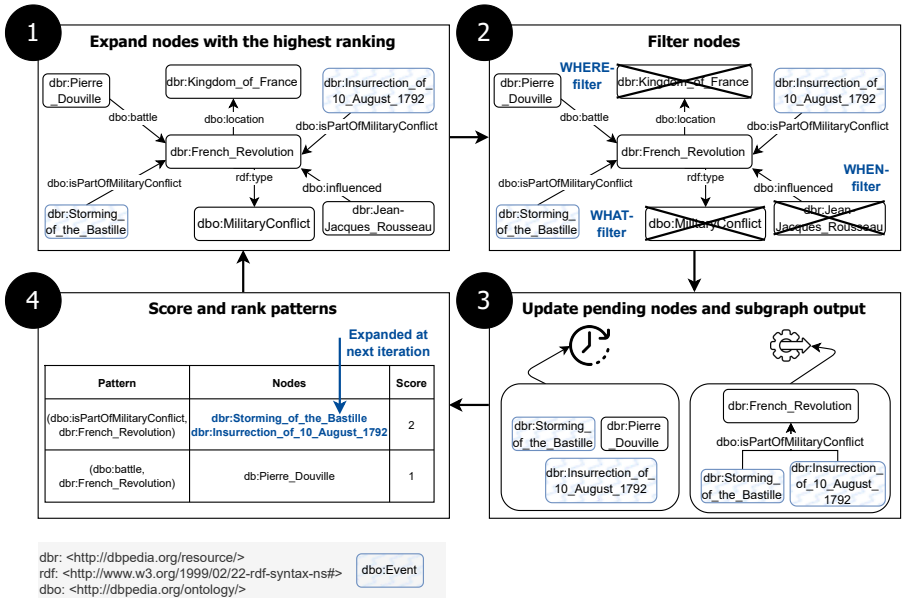

Figure 3.3: Description of one iteration during the informed graph traversal. The type of ranking is predicate-object frequency. The *WHERE*, *WHAT* and *WHEN* filters are activated (next defined as $e\_f_{\{dbo:Location\},DB,rdf:type}$, where DB is the set of triples in DBpedia, $p\_f_{\{rdf:type\}}$ and $t\_f_{\{dbo:startDate\},1789-05-05,dbo:endDate,1799-12-31}$ respectively).

To illustrate these stages more intuitively, Figure 3.3[4] shows how a single iteration of the graph search unfolds. This visual aid complements the pseudo-code by providing a concrete example of node expansion, filtering, and scoring in context. In Stage 1, the highest-ranked nodes are expanded by retrieving

[4]Images taken from `https://www.flaticon.com`.

their ingoing and outgoing neighbors. In the figure, dbr:French_Revolution is expanded. This leads to unvisited ingoing nodes like dbr:Insurrection_of_10_August_1792 or dbr:Storming_of_the_Bastille, and unvisited outgoing nodes like dbo:MilitaryConflict and dbr:Kingdom_of_France. Stage 2 applies the node filtering. For example, dbr:Kingdom_of_France is removed because it is a location node (WHERE), dbo:MilitaryConflict is discarded due to being reached via the rdf:type predicate (WHAT), and dbr:Jean-Jacques_Rousseau is excluded because he died before the French Revolution began (WHEN). These filters help focus the traversal on event nodes, which are more likely to lead to other relevant events, reducing noise during search. Importantly, some filtered nodes, typically people and locations, are re-integrated during the KG construction phase to ensure a complete and coherent event representation. Stage 3 updates the pending node set and the output subgraph with the remaining nodes. dbr:Pierre_Douville, Storming_of_the_Bastille and Insurrection_of_10_August_1792 are added to the pending nodes to possibly be expanded, and the triples connecting the two latter nodes to dbr:French_Revolution are added to the subgraph output. Lastly, in Stage 4, candidate nodes are grouped by patterns, and ranked using scoring metrics such as predicate frequency or entropy predicate frequency to guide the next iteration. In the figure, Storming_of_the_Bastille and Insurrection_of_10_August_1792 will be expanded at the next iteration.

The pseudo-algorithm and the visual overview highlight the key stages of our graph traversal process. We now formally define the event-centric filters (*Event-centric filters*) and ranking mechanism (*Ranking Mechanism*) that guide the subgraph extraction by pruning irrelevant nodes and prioritizing the most informative ones.

Event-centric filters

We define a set of generic filters to prune the search space during traversal, instantiated in our experiments as event-centric filters (see Section 3.4.1). While the filters are generic, our usage for the event-centric KGs follows the SEM ontology [338] and is aligned with its four core classes, as detailed in the introduction of Section 3.3.

The intuition behind these filters is to restrict the traversal to event-centric paths, i.e. to nodes that are themselves events. This design helps prevent the search from drifting into semantically less relevant or noisy parts of the graph. For instance, expanding a person or location node may lead to biographical or geographic information that, while informative, does not contribute directly to the sub-event structure we aim to retrieve. Therefore, such entities are, when

possible, not expanded further during traversal. However, they remain essential to the broader context of events and are reintegrated during the KG construction phase, which enriches the retrieved event-centric core with relevant contextual information such as people, places, and other linked entities.

In Figure 3.3 (Stage 2), the filters act dynamically, pruning newly visited nodes based on their type or connection. These filters could be implemented using SPARQL queries or graph navigational languages such as NautiLOD [117], though such approaches lack the ability to rank patterns. Below, we formally define the filter mechanism.

Definition 3.9 (Event-centric filters)**.** We hereafter define three types of event-centric filters: predicate-based, entity-based and temporal-based. Each filter is a boolean function that enables to prune the search space in a knowledge graph $KG = \{(s,p,o) | s \in E, o \in E \cup L, p \in R\}$.

Definition 3.9.1. ***Predicate-based filter.***
Let $R' \subset R$ and $(s,p,o) \in KG$. The predicate-based filter discards (s,p,o) from the search space if $p \in R'$. It can be formally defined as:

$$p\_f_{R'} : KG \longrightarrow \{0,1\}$$
$$(s,p,o) \longmapsto \begin{cases} 1 & \text{if } p \in R' \\ 0 & \text{else} \end{cases}$$

- **WHAT-filter.** For example, $R' = \{\texttt{rdf:type}\}$.

Definition 3.9.2. ***Entity-based filter.***
Let $E' \subset E$, $p \in R$, $s \in E$. The entity-based filter discards s from the search space if s is of any type in E'. It can be formally defined as:

$$e\_f_{E',KG,p} : E \longrightarrow \{0,1\}$$
$$s \longmapsto \begin{cases} 1 & \text{if } \exists t \in E', \text{ such that } (s,p,t) \in KG \\ 0 & \text{else} \end{cases}$$

- **WHO-filter.** For example, $E' = \{\texttt{dbo:Person}\}$, $p = \texttt{rdf:type}$.
- **WHERE-filter.** For example, $E' = \{\texttt{dbo:Location}\}$, $p = \texttt{rdf:type}$.

Definition 3.9.3. ***Temporal-based filter.***
Let $L_d \subseteq L$ be the subset of literals corresponding to dates. Let $(p_s, p_e) \in R^2$, $(t_s, t_e) \in L_d^2$. p_s and p_e are predicates denoting the start and end dates of an event, and t_s and t_e are literal timestamps. The temporal-based filter discards

$s \in E$ from the search space if it ends before t_s or if it starts after t_e. It can be formally defined as:

$$t\_f_{p_s,t_s,p_e,t_e} : E \longrightarrow \{0,1\}$$

$$s \longmapsto \begin{cases} 1 & \text{if } \exists t_1 \in L_d, \text{ such that } (s,p_s,t_1) \in KG \text{ and } t_1 > t_e \\ & \quad \text{or } \exists t_2 \in L_d, \text{ such that } (s,p_e,t_2) \in KG \text{ and } t_2 < t_s, \\ 0 & \text{otherwise.} \end{cases}$$

- **WHEN-filter.** For example, $p_s =$ `dbo:birthDate`, $p_e =$ `dbo:deathDate`, $t_s =$ "1789-05-05", $t_e =$ "1799-12-31".

The above filters are parameters of the traversal, and can be activated on demand. In Figure 3.3, the filters we activated are WHAT, WHERE, and WHEN (formally defined as $e\_f_{\{dbo:Location\},DB,rdf:type}$, $p\_f_{\{rdf:type\}}$ and $t\_f_{dbo:startDate,1789-05-05,dbo:endDate,1799-12-31}$ respectively). For WHAT, `dbo:MilitaryConflict` is discarded from the search space, since it is the object of the triple `(dbr:French_Revolution,rdf:type,dbo:MilitaryConflict)`. Regarding WHERE, `Kingdom_of_France` is discarded since it is of type `dbo:Location`. Lastly, `Jean-Jacques_Rousseau` is discarded because Rousseau died before the French Revolution started, hence applying the WHEN filter.

Ranking Mechanism

To prioritize the most relevant nodes for expansion, we implement a ranking mechanism that operates at the level of relation patterns. Rather than scoring individual nodes in isolation, our method groups nodes according to the structural and semantic patterns they instantiate in the graph. Each pattern is then evaluated and ranked based on its relevance to the current search context.

The ranking strategy combines two complementary components: a quantitative heuristic, which assigns a numerical score to each pattern, and a semantic preference function, which classifies patterns into semantically relevant vs. less semantically relevant. The final score of a pattern integrates both aspects to reflect both contextual fit and semantic importance. Once the top-ranked pattern is selected, all nodes in the pending set that match this pattern are expanded. The process repeats until no relevant patterns or nodes remain, or until the maximum iteration number is reached. We now formally introduce the notion of relation patterns and node satisfaction, before detailing the scoring and preference mechanisms used to rank them. All the examples are taken from Figure 3.3.

Definition 3.10 (Relation Patterns and Node Satisfaction)**.**
Let $KG = \{(s,p,o) | s \in E, o \in E \cup L, p \in R\}$. We define the following relation patterns in KG as Boolean functions. For each pattern defined below, we write that a node $n \in E$ **satisfies** a pattern if the corresponding pattern function returns 1 for n.

Definition 3.10.1. ***Single-step outgoing predicate pattern.***
Let a predicate $p \in R$, we formally define $opp_{p,KG}$ as follows:

$$opp_{p,KG} \colon E \longrightarrow \{0,1\}$$
$$n \longmapsto \begin{cases} 1 & \text{if } \exists o \in E \text{ such that } (n,p,o) \in KG, \\ 0 & \text{otherwise.} \end{cases}$$

- **Example.** We assume $KG = DBpedia$ and $p =$ `dbo:isPartOfMilitaryConflict`. `dbr:Storming_of_the_Bastille` satisfies $opp_{p,KG}$, since (`dbr:Storming_of_the_Bastille`, p, `dbr:French_Revolution`) $\in KG$.

Definition 3.10.2. ***Single-step ingoing predicate pattern***
Let a predicate $p \in R$, we formally define $ipp_{p,KG}$ as follows:

$$ipp_{p,KG} \colon E \longrightarrow \{0,1\}$$
$$n \longmapsto \begin{cases} 1 & \text{if } \exists s \in E \text{ such that } (s,p,n) \in KG, \\ 0 & \text{otherwise.} \end{cases}$$

- **Example.** We assume $KG = DBpedia$ and $p =$ `dbo:isPartOfMilitaryConflict`. `dbr:French_Revolution` satisfies $ipp_{p,KG}$, since (`dbr:Storming_of_the_Bastille`, p, `dbr:French_Revolution`) $\in KG$.

Definition 3.10.3. ***Single-step outgoing predicate-object pattern***
Let a predicate $p \in R$ and an entity $e \in E$, we formally define $opop_{p,e,KG}$ as follows:

$$opop_{p,e,KG} \colon E \longrightarrow \{0,1\}$$
$$n \longmapsto \begin{cases} 1 & \text{if } (n,p,e) \in KG, \\ 0 & \text{otherwise.} \end{cases}$$

- **Example.** We assume $KG = DBpedia$, $p =$ `dbo:isPartOfMilitaryConflict` and $e =$ `dbr:French_Revolution`. `dbr:Storming_of_the_Bastille` satisfies $opop_{p,e,KG}$, since (`dbr:Storming_of_the_Bastille`, p, `dbr:French_Revolution`) $\in KG$.

Definition 3.10.4. ***Single-step ingoing predicate-object pattern***
Let a predicate $p \in R$ and an entity $e \in E$, we formally define $ipop_{p,e,KG}$ as follows:

$$ipop_{p,e,KG}\colon E \longrightarrow \{0,1\}$$
$$n \longmapsto \begin{cases} 1 & \text{if } (e,p,n) \in KG, \\ 0 & \text{otherwise.} \end{cases}$$

- **Example.** We assume $KG = DBpedia$, $p =$`dbo:isPartOfMilitaryConflict` and $e =$ `dbr:Storming_of_the_Bastille`. `dbr:French_Revolution` satisfies $ipop_{p,e,KG}$, since (`dbr:Storming_of_the_Bastille`, p, `dbr:French_Revolution`) $\in KG$.

Definition 3.11 (Scoring)**.** Let G' a non-empty subset of KG.
We first define scoring metrics for ingoing and outgoing predicate patterns in G'. Let a predicate $p \in R$ and the patterns ipp_p and opp_p from Definition 3.10.

- Predicate Frequency *(pf).* Edges with predicates that are often used in the dataset will be favoured.

$$pf(ipp_p, G') = |\{s \mid \exists o \in E, \text{ such that } (s,p,o) \in G'\}|$$

$$pf(opp_p, G') = |\{o \mid \exists s \in E, \text{ such that } (s,p,o) \in G'\}|$$

- Entropy Predicate Frequency *(epf).* Edges with the most informative predicates in the dataset will be favoured. Let $pp \in \{ipp_p, opp_p\}$.

$$epf(pp, G') = \begin{cases} -\frac{pf(pp,G')}{|G'|} \cdot \log(\frac{pf(pp,G')}{|G'|}) & \text{if } pf(pp,G') > 0, \\ 0 & \text{otherwise.} \end{cases}$$

- Inverse Predicate Frequency *(ipf).* Edges with predicates that are rarely used in the dataset will be favoured. Let $pp \in \{ipp_p, opp_p\}$.

$$epf(pp, G') = \begin{cases} -\frac{1}{pf(pp,G')} & \text{if } pf(pp,G') > 0, \\ 0 & \text{otherwise.} \end{cases}$$

We second define scoring metrics for ingoing and outgoing predicate-object patterns. Let p, $p \in R$, e, $e \in E$, and the patterns $ipop_{p,e}$ and $opop_{p,e}$ from Definition 3.10.

- Predicate-Object Frequency *(pof)*. This is similar to *pf*, but this metrics also differentiates between the objects.

$$pof(ipop_{p,e}, G') = |\{s \mid (s,p,e) \in G'\}|$$
$$pof(opop_{p,e}, G') = |\{o \mid (e,p,o) \in G'\}|$$

- Entropy Predicate-Object Frequency *(epof)*. This is similar to *epf*, but this metrics also differentiates between the objects.
  Let $pop \in \{ipop_{p,e}, opop_{p,e}\}$.

$$epof(pop, G') = \begin{cases} -\frac{pof(pop,G')}{|G'|} \cdot \log(\frac{pof(pop,G')}{|G'|}) & \text{if } pof(pop,G') > 0, \\ 0 & \text{otherwise.} \end{cases}$$

- Inverse Predicate-Object Frequency *(ipof)*. This is similar to *ipf*, but this metrics also differentiates between the objects.
  Let $pop \in \{ipop_{p,e}, opop_{p,e}\}$.

$$ipof(pop, G') = \begin{cases} \frac{1}{pof(pop,G')} & \text{if } pof(pop,G') > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Let us consider the following triples from Table 3.3. This ensemble of triples is denoted G' below. We have $|G'| = 3$.

Table 3.3: Triples considered for the scoring example. Only unvisited nodes are taken into account.

Subject	Predicate	Object
dbr:Pierre_Douville	dbo:battle	dbr:French_Revolution
dbr:Insurrection_of_10_August_1792	dbo:isPartOfMilitaryConflict	dbr:French_Revolution
dbr:Storming_of_the_Bastille	dbo:isPartOfMilitaryConflict	dbr:French_Revolution

Since `dbr:French_Revolution` was already visited, we only consider the ingoing patterns. Given the above definitions, we have the followings scores:

- $pf(ipp_{\texttt{dbo:battle}}, G') = 1$
- $pf(ipp_{\texttt{dbo:isPartOfMilitaryConflict}}, G') = 2$
-

$$\begin{aligned} epf_{G'}(ipp_{\texttt{dbo:battle}}) &= -\frac{pf(ipp_{\texttt{dbo:battle}}, G')}{|G'|} \cdot \log\left(\frac{pf(ipp_{\texttt{dbo:battle}}, G')}{|G'|}\right) \\ &= -\tfrac{1}{3} \cdot \log\left(\tfrac{1}{3}\right) \\ &\approx 0.16 \end{aligned}$$

- We note `dbo:isPartOfMilitaryConflict` as `dbo:iPOMC` below for more clarity.

$$\begin{aligned} epf\left(ipp_{\text{dbo:iPOMC}}, G'\right) &= -\frac{pf(ipp_{\text{dbo:iPOMC}})}{,} G'|G'| \cdot \log\left(\frac{pf(ipp_{\text{dbo:iPOMC}}, G')}{|G'|}\right) \\ &= -\tfrac{2}{3} \cdot \log\left(\tfrac{2}{3}\right) \\ &\approx 0.12 \end{aligned}$$

The scoring metric used in Figure 3.3 is *pof*. The score of the pattern $ipop_{\texttt{dbo:isPartOfMilitaryConflict,dbr:French\_Revolution}}$ is 2 since there are two ingoing nodes of the `dbr:French_Revolution` that are connected to the latter with the predicate `dbo:isPartOfMilitaryConflict`. If the scoring metric was *epof*, the score would be $-\frac{2}{3} \cdot \log \frac{2}{3} \approx 0.12$.

Definition 3.12 (Preference Function)**.** The preference function is a boolean function that classifies a pattern as semantically relevant if it meets certain conditions. Let $KG = \{(s,p,o) | s \in E, o \in E \cup L, p \in R\}$ a KG. Let $d \in R$ and $sc \in R$ denoting predicates for domain and subclass information. Let $extract\_type \in E$ a type of events to retrieve. Let $p \in R$, $e \in E$ and a pattern pp, $pp \in \{ipp_p, opp_p, ipop_{p,e}, opop_{p,e}\}$. We formally define:

$$pref_{d,sc,t\_e}(pp) = \begin{cases} 1 & \text{if } \exists n \in E, \text{ such that } (p,d,n) \in KG \\ & \text{and} (n, sc, extract\_type) \in KG \\ 0 & \text{otherwise.} \end{cases}$$

- For example, $d =$ `rdfs:domain`, $sc =$ `rdfs:subClassOf`, $t\_e =$ `dbo:Event`.

The preference function prioritises predicates with the highest scores from the *scoring* step, favouring nodes with the correct types. We use the ontological equivalent predicates of `rdfs:domain` and `rdfs:range`. In DBpedia, we have `(dbo:isPartOf MilitaryConflict,rdfs:domain,dbo:MilitaryConflict)` and `(dbo: MilitaryConflict,rdfs:subClassOf*,dbo:Event)`. In `(s,dbo:isPartOfMilitaryConflict,o)`, s must be of type `dbo:Event`. With the preference function, the system prefers patterns with `dbo:isPartOfMilitaryConflict` over `dbo:battle` since the former has `dbo:MilitaryConflict` as its domain, a subclass of event type, while the latter lacks explicit information about its domain or range. In Figure 3.3, the preference function does not affect the selection of nodes to expand. Without the preference function, patterns are ranked in descending order of scores from the *scoring* step.

We have presented the event-centric subgraph extraction component of ChronoGrapher. We now present its event-centric KG population component.

3.3.2 Event-centric KG Population (RQ2)

To obtain a comprehensive, fine-grained event representation, ChronoGrapher adds a second component for building event-centric KGs. This process complements the subgraph extraction phase by enriching the retrieved events with contextual information. Such information may be explicitly encoded in the KG through IRIs or expressed in texts stored as literals.

Algorithm 2 outlines the KG construction procedure. Starting from the events identified during the subgraph extraction step, the system applies two complementary triple enrichment strategies, one structure-based and one text-based, to build the event-centric KG:

1: Structured triple enrichment (IRI-based). For each event, the system retrieves its outgoing triples from the original KG and maps them to SEM-compatible relations. This mapping relies on a predefined set of predicate labels (S in the pseudo-code), which determine the corresponding narrative dimensions (e.g., time, place, actor). This step is rule-based and aims to reproduce the type of structured information found in EventKG [135].

2: Text-based triple enrichment (abstract-based). The system then processes the abstract associated with each event, typically encoded as a literal. Using a semantic information extraction pipeline based on frame semantics, it identifies frames and their roles (frame elements), which are then linked to named entities. This allows the construction of more fine-grained relations between events and entities, such as causal relations.

An example of the IRI-based KG construction is shown in Figure 3.4. On the left, we show a portion of the original Wikidata subgraph for the Coup of 18 Brumaire (`wd:Q620965`), while the right side presents the transformed, SEM-aligned event-centric KG. Table 3.4 lists the label patterns used in this process. For each triple `(s, p, o)`, if the label of `p` matches one of the predefined labels associated with a narrative dimension (e.g., time, place, actor), a corresponding SEM-compatible triple is added to the output graph. This transformation is governed by a rule-based mapping approach, which assigns SEM predicates based on the labels of the original RDF predicates. For instance, consider a predicate whose label contains the substring *"person"*. This label indicates a relation involving an actor. In this case, the system adds the triple `(s, sem:hasActor, o)` to the event-centric KG.[5]

[5]The rules are implemented in the file `src/build_ng/generic_kb_to_ng.py`.

Algorithm 2 Event-centric KG Population

Input:

- $KG = \{(s,p,o) \mid s \in E, o \in E \cup L, p \in R\}$: knowledge graph to traverse;
- G': output of Algorithm 1;
- n_{start}: starting node for the search;
- t_s, t_e: start and end dates of n_{start}.

Output: A: event-centric KG

$S \leftarrow \{s_1, ..., s_n\}$ # Predicate labels to keep, predefined (for the KG construction)
$A \leftarrow \emptyset$
$E \leftarrow$ GETEVENTS$(G') \cup \{\text{n}_{start}\}$ # Retrieve list of events from G'
for all $e \in E$ **do**
  $A \leftarrow A \cup \{(e, \text{rdf:type}, \text{sem:Event})\}$ # Update output KG with event
  $outgoing \leftarrow$ OUTGOING(e, KG) # 1. Structured triple enrichment (IRI-based)
  for all $(e,p,o) \in outgoing$ **do**
    $label \leftarrow$ GETLABEL(p)
    if $p \in S$ **then**
      $A \leftarrow A \cup$ GETSEMTRIPLE$(e,p,o,n_{start},t_s,t_e)$ # This step may not always add triples
    end if
  end for
  $abstract \leftarrow$ GETABSTRACT$(event)$ # 2. Text-based triple enrichment (abstract-based)
  $frames \leftarrow$ GETFRAME$(abstract)$
  for all $f \in frames$ **do**
    $A \leftarrow A \cup$ GETFRAMEKG(f)
  end for
end for
return A

Beyond transforming IRI triples, ChronoGrapher enriches the KG by extracting structured information from textual description, specifically, event abstracts encoded as literals. To achieve this, we rely on the frame semantics theory [109], which represents situations as frames, i.e. as structured templates involving participants, temporal properties, and other contextual roles.

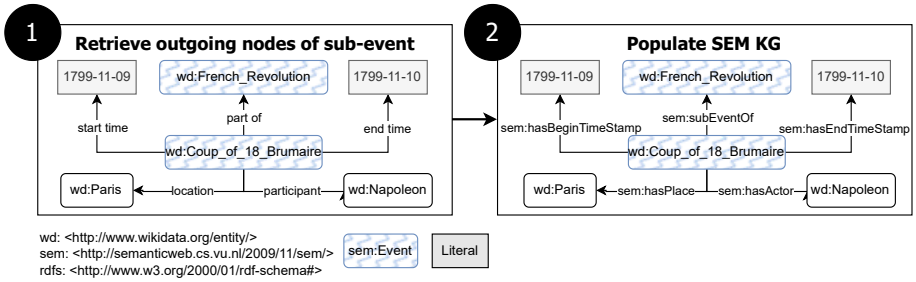

Figure 3.4: Populating a SEM KG from the Coup of 18 Brumaire in Wikidata. *wd* is a prefix used in Wikidata for `http://www.wikidata.org/entity/`.

Table 3.4: Labels used to retrieve information from KG triples.

Narrative dimension	SEM predicate	Labels
Who	`sem:hasActor`	person, combatant, commander, participant
When (begin)	`sem:hasBeginTimeStamp`	start time, date, point in time
When (end)	`sem:hasEndTimeStamp`	end time
Where	`sem:hasPlace`	place, location, country
Part of	`sem:subEventOf`	partof, part of
Part of (inverse)	`sem:hasSubEvent`	has part, significant event

Some event-centric KGs built from news articles [204, 279] also used frames. Although EventKG contains text events and links them to entities, it does not provide the finer-grained semantic structure of frames. One example of frames is the `Change_of_ leadership` frame[6]. It describes "the appointment of a `New_leader` or removal from office of an `Old_leader`". Semantic roles such as `Old_leader` and `New_leader` are called frame elements.

ChronoGrapher automatically identifies such frames and their associated roles by reusing the pre-trained transformer-based model introduced by [61]. Each frame instance is aligned with the NIF ontology [153] to anchor it in the source text, and DBpedia Spotlight [237] is used to link frame elements to entities. The resulting frame structures are then integrated into the KG and linked to the Framester ontology [122], using a representation consistent with Framester's semantic model[7][8].

[6] `https://framenet2.icsi.berkeley.edu/fnReports/data/frameIndex.xml?frame=Change_of_leadership`

[7] `http://etna.istc.cnr.it/framesterpage/wsj/wsjpropnetannotations/CE_55094`.

[8] The code for this part is in the `src/build_ng/frame_semantic.py` file.

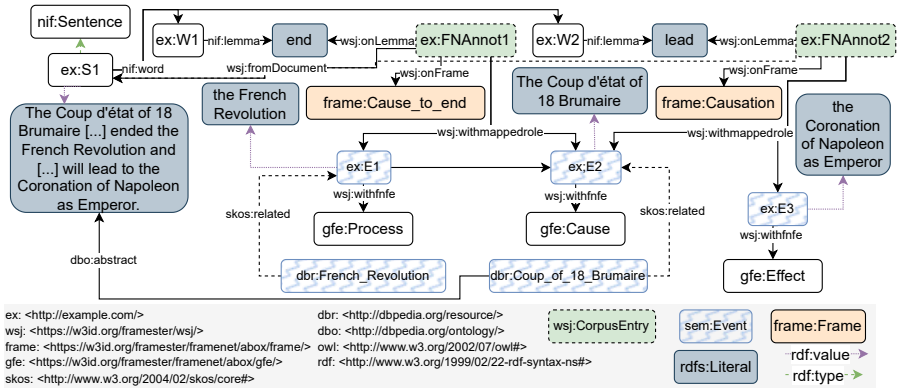

Figure 3.5: Example of frames extracted from text about the *Coup of 18 Brumaire*. The KG contains both literals (in grey) and linked KG entities. For instance, the French Revolution appears as a literal, directly tied to its sentence and serving as the value of role `ex:E1`, while the linked DBpedia entity (`dbr:French_Revolution`) is associated to `ex:E1` via `skos:related`.

Figure 3.5 illustrates this process. Starting from the abstract of `dbr:Coup_of_18_Brumaire`, the text is segmented into sentences (e.g., `ex:S1`). A frame annotation such as `ex:FNAnnot1` is associated with the frame `frame:Cause_to_end`, which in turn connects to roles like `ex:E1`, representing entities such as the French Revolution. These literals are grounded to IRI entities (e.g., `dbr:French_Revolution`) via entity linking. This frame-based enrichment enables ChronoGrapher to capture causal, temporal, and participant-related semantics that go beyond what is available from structured triples alone, and is particularly useful for answering complex event-centric queries (as explored in RQ3).

3.4 Evaluation

We present the setup (Section 3.4.1), results on the quality of the method (Section 3.4.2) as well as the impact of the constructed event-centric KGs on LLMs question-answering (Section 3.4.3). The quantitative evaluation (Section 3.4.2, RQ1 and RQ2) focuses on evaluating the quality of the method, both for the traversal (RQ1) and the KG population (RQ2). The evaluation for RQ1 is done automatically and can be reproduced. The evaluation for RQ2 is partly done automatically, and partly manually assessed. For RQ3, we conducted a preliminary user study to assess the potential of event-centric KGs on a LLM QA setting, and we found encouraging qualitative results.

3.4.1 Experimental Setting (RQ1, RQ2)

Our experiments first aim to assess the quality of ChronoGrapher, both for event-centric subgraph extraction (RQ1) and for event-centric KG population (RQ2). For RQ1, we investigate the effects of filtering and ranking on the graph traversal approach. We first experiment under different configurations, with six distinct parameters for the search process: (1) scoring metric, (2) preference function (cf. Section 3.3.1), (3-6) filters pertaining to WHAT, WHERE, WHEN, and WHO (Definition 3.9). The scoring metrics is one of the six options described above (Definition 3.11), whereas all the other parameters are boolean.

To speed up the overall experiments, we used two Ubuntu machines with similar hardware configurations (40 CPU cores and 314GiB / 376GiB of memory, respectively). Given the close similarity in specs, we do not expect the use of different machines to have introduced any meaningful variability in performance. While memory usage remains relatively constant due to the persistent in-memory KG, the approach would be more CPU-intensive, as it involves issuing a large number of query interface calls during execution.

Parameter Notation. Throughout this section, we use the following binary parameters to indicate which semantic filters or retrieval strategies are enabled in each configuration:

- `what`, `where`, `when`, `who` $\in \{0,1\}$: These indicate whether the corresponding semantic filters are active. For instance, `who` = 1 means the WHO-filter defined in Section 3.3.1 is applied.
- `domain_range` $\in \{0,1\}$: When set to 1, the domain-range preference function is used to prioritize triples whose predicate-object structure aligns with the expected answer type.

Baselines. For RQ1 on event-centric subgraph extraction, we compare our system *chronographer* with three categories of system we implemented. We adapted LDSpider [170] and NautiLOD [117] to align with our method. The performance comparison between *chronographer* and the considered baselines focuses solely on the quality of the extracted sub-events, not the entire subgraph.

- *random-5*, *random-10* and *random-15*: at each iteration, *nb* nodes are randomly selected to be expanded. We used $nb \in \{5, 10, 15\}$. Parameters (1-6) are non applicable.

- *ldspider-m*: inspired by LDSpider [170], this search is a breadth-first search with some predefined predicates to be pruned out of the search space. Parameters (1) and (2) are not applicable, and (3-6) are set to 0.

- *nautilod-m*: inspired by NautiLOD [117], this baseline works as a breadth-first search combined with more complex filters than *ldspider-m*. Parameters (1) and (2) are not applicable, but we experiment with (3-6).

Evaluation. We chose EventKG 3.1 [133] as a golden standard primarily because, to the best of our knowledge, it is the only existing resource that automatically constructs event-centric knowledge graphs from large-scale KGs like DBpedia and Wikidata, making it the most relevant point of comparison for our method. Additionally, EventKG relies on similar input datasets, which allows for a meaningful overlap in evaluation. While we note that the original EventKG paper does not detail a formal evaluation of its own event selection process, its scale, structured format, and availability make it a valuable reference baseline for both subgraph extraction and KG construction tasks. We discuss this further in Section 3.5.1.

For RQ1 on event-centric subgraph extraction, we constructed a gold standard by querying EventKG using SPARQL SELECT queries to retrieve relevant event instances. The notebook is available to replicate this process[9]. For RQ2 on event-centric KG population, we used SPARQL CONSTRUCT queries to adapt EventKG's triples, specifically, we replaced EventKG IRIs with the original DBpedia/Wikidata instances and aligned the predicates to SEM. The code implementing this adaptation is publicly available[10].

We chose to compare our method's output to EventKG using the standard F_1, precision and recall metrics. For RQ1, which focuses on extracting a subgraph from a KG given an input event, we use these metrics at the event level to evaluate whether the extracted subject or object nodes are indeed sub-events of the target event. This allows us to assess the correctness and completeness of the events in the subgraph. For RQ2, which concerns constructing an event-centric KG, we use the same metrics at the triple level to evaluate whether the resulting KG correctly describes sub-events with relevant features like place, time, or actor. This helps us determine how well the structural component of the KG captures the intended event semantics. These metrics thus offer a standard and interpretable way to assess performance for both subgraph extraction and KG construction.

[9] `https://github.com/SonyCSLParis/graph_search_framework/blob/main/notebooks/eventkg-retrieving-events.ipynb`

[10] `https://github.com/SonyCSLParis/graph_search_framework/blob/main/src/build_ng/eventkg_to_ng.py`

Datasets. Our experiments focus on DBpedia (2021-09) [15], Wikidata (2021-03-05) [344] and YAGO4 [260] because EventKG was built on them. We use the HDT (Header, Dictionary, Triples) version of DBpedia, which we obtained from TriplyDB[11]. We obtain Wikidata from rdfhdt[12] and YAGO4 from their website[13]. We chose HDT primarily for performance and scalability reasons when working with large datasets such as DBpedia and Wikidata. HDT allows for fast, local querying thanks to its compressed, indexed structure, which significantly reduces loading and access times compared to traditional RDF formats or remote SPARQL endpoints[14]. This is especially suitable for tasks involving repeated and large-scale traversals, such as our method. However, our method is not tied to HDT. We also implemented a SPARQL-based interface, enabling the traversal to be executed on any SPARQL endpoint. This ensures the approach remains flexible and broadly applicable beyond the HDT setting.

Table 3.5 shows information on events and their sub-events across datasets. All types of events are taken into account, which mainly include historical events, sports events and political events such as elections. We choose to retain solely the historical events with more than ten sub-events, as they offer more than a simple list of events. Nonetheless, we provide results on additional experiments on sports and political events in Section 3.5.3. The final distribution of historical events in EventKG and their average number of outgoing links are presented in Table 3.6. Events with a higher number of sub-events tend to have a higher average number of features. The limited number of outgoing links for YAGO4 resulted in traversals that would end after one iteration. YAGO4 was built on Wikidata with an improved schema, without the full content from Wikidata, and we found that the ingoing links were mostly about creative works, and the outgoing links about metadata, discarded for the search. This resulted in a limited number of features. We thus only used Wikidata and DBpedia.

Filters and preference function definition. The filters from Definition 3.9 were generic with event-centric examples for the WHAT, WHERE, WHEN and WHO filters. Table 3.7 describes the predicates we used for each of the filters. Each filter relates to one of the core classes of the SEM ontology [338], making them event-centric. However, other filters can also be implemented by instantiating other variables, making the method easily expandable and usable for other tasks.

[11] `https://triply.cc`
[12] `https://www.rdfhdt.org/datasets/`
[13] `https://yago-knowledge.org/downloads/yago-4`
[14] `https://www.rdfhdt.org/what-is-hdt/`

Table 3.5: Statistics on events and their sub-events across datasets, extracted from EventKG. Each cell shows, per dataset, the number of events with at least one sub-event, broken down by the total number of sub-events: exactly one (=1), more than one (>1), and more than ten (>10). The "Retained" column indicates the subset of historical events with more than ten sub-events that were retained for our experiments.

Dataset	Nb. of sub-events			
	= 1	>1	>10	Retained
Wikidata	203,988	238,094	2,408	341
DBpedia	84,599	95,504	1,333	250
YAGO4	70,738	76,682	993	306

Table 3.6: For each dataset, the left block reports how many events contain more than 10, 20, 30, 50, or 100 sub-events, and the right block reports the average number of features of with those events for the same thresholds. Except for YAGO4, events with more sub-events have more features on average.

Dataset	Nb. of sub-events					Avg. nb. of features				
	>10	>20	>30	>50	>100	>10	>20	>30	>50	>100
Wikidata	341	208	147	91	47	237	273	290	323	381
DBpedia	275	158	106	66	37	122	133	141	147	160
YAGO4	306	180	126	76	44	5	4	5	5	5

3.4.2 Quantitative Evaluation (RQ1, RQ2)

We now present the quantitative evaluation of our method, structured around RQ1 on event-centric subgraph extraction and RQ2 on KG construction. *Parameter selection for the search (RQ1)* presents the parameter selection procedure for the event-centric subgraphs (RQ1), whereas *Search Results (RQ1)* reports the results of the search component, including comparisons with baselines. Lastly, *Event-centric KG Population Results* addresses RQ2 by evaluating the construction of enriched event-centric knowledge graphs.

Parameter selection for the search (RQ1).

For ***chronographer*** and ***nautilod-m***, we first experimented on 12 events to select the most meaningful parameters to run the graph traversal on all events. Table 3.8 shows the 12 events used for parameter selection for the search. There were 192 possible combinations of parameters ($6 \cdot 2^5$) for ***chronographer***, and 16 (2^4) for ***nautilod-m***.

Table 3.7: Instantiated variables used for each of the filters and the preference function. DB: DBpedia, WD: Wikidata.

Name	Type	Dataset	Variable	Predicates
WHAT	predicate filter	DB	R'	*rdf:type*
		WD		*wp:P31*
WHO	entity filter	DB	E'	*dbo:Person*
			p	*rdf:type*
		WD	E'	*wd:Q5*
			p	*wd:P31*
WHERE	entity filter	DB	E'	*dbo:Place*
				dbo:Location
			p	*rdf:type*
		WD	E'	*wd:P17*
				wp:P276
				wd:Q6256
			p	*wp:P31*
WHEN	temporal filter	DB	*ps*	*dbp:date*
				dbp:startDate
				dbp:birthDate
			pe	*dbp:endDate*
				dbp:deathDate
		WD	*ps*	*wp:P585*
				wp:P580
				wp:P569
			pe	*wp:P582*
				wp:P570
Preference Function	-	DB	d	*rdfs:domain*
			sc	*rdfs:subClassOf*
			t_e	*dbo:Event*
		WD	d	*wd:Q21503250*
			sc	*wp:P279*
			t_e	*wd:Q13418847*

Table 3.8: Events used for parameter selection.

World War I	American Indian Wars	Mediterranean and Middle East Theatre of World War II
French Revolution	Cold War	European Theatre of World War II
Napoleonic Wars	Coalition Wars	European Theatre of World War I
Pacific War	Russian Civil War	Yugoslav Wars

We investigated on the correlation between our ranking and filtering parameters and the best F_1 score. For ***chronographer***, we found a strong positive correlation between *domain_range* and higher F_1 scores. Prioritizing events based on their domain range thus significantly improves event retrieval performance. We observed a weak but statistically significant correlation between *when* and the F_1 score, suggesting that adding temporal information could help the search process. For ***nautilod-m***, we found a strong positive correlation between *what* and the best F_1 score. The *what* filter indeed enables to remove more generic nodes from the search, which proves its efficiency here.

Table 3.9 shows the contributions of each filter individually. As an illustration, $what = 1$ means that only the experiments with this parameter activated, and all the others set to 0, are taken into account. For ***nautilod-m***, the parameter that yields the best result is *what*, with an average F_1 of 0.36. The best average F_1 score is obtained when all parameters are activated. On average, ***chronographer*** performs better or comparably to ***nautilod-m***, under all configurations. The parameter that yields the best results is *domain_range*. For computational and performance reasons, we set the following parameters: **domain_range = when = what = where = who = 1**. We kept all metrics but the inverse ones for ***chronographer*** for the full experiments.

Lastly, Table 3.10 shows the results of all systems. We set a timeout of 10 hours for each experiment. Only 5 experiments from ***ldspider-m*** timed out. In these cases, we took the latest output of the search at that stage. Our systems achieve better result than the baselines. The search is furthermore more localised and efficient, since the output graphs and the runtime are smaller.

Search Results (RQ1)

Table 3.11 shows the final results on the remaining 604 events for all systems. The three metrics tend to improve when moving from the ***random*** baselines to our ***chronographer*** methods. ***random-5*** has the worst F_1 scores, and ***chronographer-epof*** has the best ones. The differences are the most visible in the Recall scores, which means that the ***informed*** systems leave out less false negative than the random baselines. Precision scores for Wikidata tend to be higher

Table 3.9: Mean F_1 score for individual parameters. *no filter* and *all* means that all parameters are set to 0 and 1 respectively. *pf*, *pof*, *epf*, *epof*, *ipf* and *ipf* are the six possible scoring metrics (Section 3.4.1).

	nautilod-m	*chronographer*					
		pf	*pof*	*epf*	*epof*	*ipf*	*ipof*
domain_range = 1	N/A	0.66	**0.77**	0.66	**0.77**	0.66	0.65
what = 1	0.36	0.50	0.48	0.37	0.39	0.25	0.27
where = 1	0.28	0.47	0.48	0.33	0.39	0.25	0.23
when = 1	0.27	0.47	0.48	0.45	0.43	0.29	0.30
who = 1	0.25	0.43	0.50	0.39	0.48	0.26	0.26
no filter	0.31	0.46	0.48	0.34	0.39	0.24	0.24
all	0.53	0.69	**0.77**	0.69	**0.77**	0.69	0.72

Table 3.10: Results comparison of various systems.

System	F_1	Precision	Recall	Nb_Nodes	Nb_Preds	Nb_Triples	Runtime
random-5	0.24	0.68	0.16	40,356	70	40,915	38'00"
random-10	0.31	0.54	0.24	1,554	70	**2,027**	43'00"
random-15	0.28	0.57	0.21	11,890	89	12,744	25'00"
ldspider-m	0.23	0.70	0.23	126,015	137	241,049	23'00"
nautilod-m	0.53	0.51	0.63	5,096	121	13,755	5'20"
chronographer-pf	0.69	0.70	0.71	874	21	2,366	1'51"
chronographer-pof	**0.78**	**0.79**	0.77	750	**17**	2,090	2'13"
chronographer-epf	0.69	0.70	0.72	857	20	2,353	**1'35"**
chronographer-epof	**0.77**	0.78	**0.77**	**726**	**17**	2,082	1'42"

than the ones for DBpedia, whereas it is the reverse for Recall. Therefore, the content retrieved in Wikidata is less noisy but also less complete. The ***random*** baselines and ***ldspider-m*** yield significantly bigger graphs than ***nautilod-m*** and the ***chronographer*** baselines, and take more time to run. ***nautilod-m*** performs better than the above baselines, however the size of the graphs and the runtime remain higher than the ***chronographer*** baselines (·4).

To summarise findings on event-centric subgraph extraction (RQ1), adding event-centric filters (WHAT, WHERE, WHEN, WHO) and a ranking step with an entropy predicate-object frequency heuristics helps to select relevant events while being more efficient time-wise and while exploring smaller parts of the graphs, as shown by the better performance of the ***chronographer*** systems in Table 3.11.

Table 3.11: Systems results on the 604 events. *DB*: DBpedia, *WD*: Wikidata.

System	F_1		Precision		Recall		Nb Nodes		Nb Preds		Nb Triples		Runtime	
	DB	WD	DB	WD	DB	WD	DB	WD	DB	WD	DB	WD	DB	WD
random-5	0.65	0.63	0.72	0.79	0.68	0.59	22,185	427	23	9	23,221	530	23'00"	11'00"
random-10	0.65	0.64	0.72	0.78	0.69	0.60	30,078	1,033	27	10	33,264	1,470	18'56"	11'33"
random-15	0.64	0.63	0.71	0.78	0.68	0.60	41,868	1,139	27	11	53,561	1,582	18'26"	16'43"
ldspider-m	0.64	0.63	0.73	0.78	0.68	0.61	131,607	4,992	52	20	239,548	7,389	27'05"	44'05"
nautilod-m	0.70	0.66	0.72	0.81	0.74	0.62	481	4,787	23	14	1,150	7,321	2'02"	39'26"
chronographer-pf	0.75	0.67	0.75	0.82	0.80	0.63	108	661	**5**	**6**	213	1,146	26"	9'38"
chronographer-pof	0.77	**0.68**	0.76	**0.83**	**0.82**	**0.64**	**97**	**612**	**5**	**6**	**196**	**863**	25"	8'43"
chronographer-epf	0.75	0.67	0.75	**0.83**	0.80	0.62	108	670	6	**6**	212	963	32"	9'07"
chronographer-epof	**0.78**	**0.68**	**0.77**	**0.83**	**0.82**	0.63	99	752	**5**	**6**	199	1,093	**24"**	**8'16"**

Event-centric KG Population Results (RQ2)

To generate our event-centric KGs, we use information from triples (i) and text (ii). On average, over all 281 historical events with more than 10 sub-events from DBpedia, there were 28,123 triples generated during the extraction from text, and 548 other triples. There are therefore 51 more triples generated during the extraction from text than other triples. For the triples generated from the information extraction the KG predicates are distributed, on average per event, as: 41.9% *rdf* predicates, 40.0% *wsj* predicates, 12.2% *nif* predicates, 5.6% *skos* predicates, and 0.3% *dbo:abstract* predicates.

(i) Constructing event-centric KG from triples. We generate event-centric KGs for each event using all sub-events from the ground truth. We compare the number of common triples against EventKG. The results are shown in Table 3.12. ChronoGrapher achieves an overall F_1 score of 67.2% and 76% for DBpedia and Wikidata. Precision scores for DBpedia are higher than for Wikidata, which means that the event-centric KGs are less noisy but less complete. For Wikidata, it is the opposite.

Table 3.12: Our event-centric KGs against EventKG. *DB*: DBpedia, *WD*: Wikidata.

Predicate	F_1		Precision		Recall	
	DB	WD	DB	WD	DB	WD
`all`	67.2	76.0	92.1	72.1	52.9	80.4
`sem:hasActor`	65.0	43.6	90.4	58.1	50.7	34.9
`sem:hasBeginTimeStamp`	77.7	81.9	90.5	71.7	68.0	95.5
`sem:hasEndTimeStamp`	77.7	75.9	90.5	70.4	68.1	82.4
`sem:hasPlace`	63.6	90.3	95.0	92.1	47.8	88.6

Table 3.13 shows the F_1, Precision and Recall scores of the narrative graphs generated from the output of the search algorithm presented in Section 3.3.1. Our end-to-end system achieves an overall (for all predicates) F_1 score of 51.7% and 49.2% respectively. As in Table 3.12, precision is higher for DBpedia, while recall tends to be higher for Wikidata. The results are furthermore lower than those in Table 3.12, which is expected since the output of the search also contains events that are not in the ground truth events. Consequently, for each of such event, all generated triples will not be in the ground truth from EventKG.

Table 3.13: Metrics of generated narrative graphs compared to EventKG.

Pred	F_1		Precision		Recall	
	DB	WD	DB	WD	DB	WD
`all`	51.7	49.2	79.3	41.1	38.4	61.4
`sem:hasActor`	48.5	27.6	76.5	35.1	35.5	22.8
`sem:hasBeginTimeStamp`	62.6	50.3	79.5	38.0	51.6	74.4
`sem:hasEndTimeStamp`	62.7	48.3	79.5	38.1	51.7	65.9
`sem:hasPlace`	48.2	59.2	81.6	50.7	34.2	71.2

(ii) Constructing event-centric KG from text. We generate event-centric KGs from all the DBpedia events and sub-events that contain an abstract. The content that is extracted in this part is not present in EventKG. In total, this sums up to 9,438 events. On average, each event had 18 distinct frames and 48 instantiations of frames with at least one mapped role. The frames that appear the most in terms of number are: `Military` (26,681), `Hostile_encounter` (26,111), `Political_locales` (17,513), `Attack` (16,155), and `Leadership` (14,002). We are interested in events and relations between events, hence we focus on "causation" frames for a qualitative analysis, but many other frames are extracted. We manually annotate (1) whether there is a causation frame (2) whether the cause and the effect are correctly identified (3) whether the DBpedia entity is correct for 100 frame annotations. Out of 100 frames, 97 had at least one identified Cause or Effect. 85 were correctly identified as Causation frames. False positive causation frames often contain causative lemmas such as the noun "cause" that might have induced the model in error. Among the 171 retrieved frame elements, 151 (88.3%) were correct. The majority of wrong frame elements resulted from false positive causation frames. Among the 94 DBpedia entities that were identified, 80 (85.1%) were correct.

To summarise findings on event-centric KG population (RQ2), we use both information extraction from triples and text. For triples, our system achieves

F_1 scores of 67.2% and 76% for DBpedia and Wikidata. For text, we conduct a qualitative evaluation on the "causation" frames and find that the system is able to retrieve information with good performance.

To illustrate the constructed KGs, we provide several sample KG files in the public repository[15], along with a file describing the contents of each. These examples, based on the French Revolution scenario, illustrate the different KG construction strategies used in our pipeline. Specifically, `eventkg_ng.ttl` corresponds to the KG generated from EventKG; `frame_ng.ttl` is derived from event abstracts (abstract-based); `generation_ng.ttl` results from the ground-truth event IRIs (IRI-based); and `search_ng.ttl` captures the KG generated from the output of the subgraph extraction component (IRI-based).

3.4.3 User Study (RQ3)

We conducted a user study focusing on a QA task. The goal of the study was to evaluate whether integrating our constructed event-centric KGs could enhance the quality of answers produced by LLMs, compared to using LLMs alone. *User Study Setup* first outlines the design of the user study, including the preparation of evaluation data and the experimental setup. *User Study Results* presents and analyzes the results of our user study.

User Study Setup

To evaluate whether integrating event-centric KGs enhances the performance of LLMs compared to using LLMs alone (RQ3), we designed 6 SEM-centric question types, each reflecting a core role or relation from the SEM ontology: summary of an event, causal explanations for an event, types of events within a time-range, sub-events of an event, events in which an actor participated, and events in which two actors both participated. For each type, we created a question template and instantiated it with two manually selected events, resulting in 12 questions. Each question was manually associated with an event during the design phase, no automated mapping was used. Once the event was selected, we retrieved its corresponding event-centric KG using SPARQL queries, and used the resulting triples as context in the LLM prompt alongside the natural language question. Table 3.14 shows examples of the 6 types of questions.

We used three different types of prompts: (a) `base`: base prompt, without context triples, (b) `db-kg`: `base` + context triples from DBpedia, and (c) `ec-kg`: `base` + context triples from the event-centric KG built by ChronoGrapher. Table 3.15 details how context triples are retrieved for each question type,

[15] `https://github.com/SonyCSLParis/graph_search_framework/tree/main/kg-example`

Table 3.14: Event-centric templated question examples.

#	Type	Example
1	Summary	*"Please provide a summary of the French Revolution"*
2	Causal explanation	*"What happened at the end of the French Revolution? Please provide the main causal happenings that led to the unfolding of the French Revolution."*
3	Event types within a time range	*"What were the main type of events that happened between 1792-01-01 and 1793-01-01 during the French Revolution?"*
4	Sub-events	*"What were the main events of the French Revolutionary Wars during the French Revolution? Can you list and order them in chronological order?"*
5	One actor in an event	*"What happened to Antoine Balland during the French Revolution?"*
6	Two actors in an event	*"In which events were Jean-Baptiste Jourdan and Joseph Bonaparte both involved during the French Revolution?"*

both for DBpedia and ChronoGrapher. The detailed queries are available in the code[16]. To integrate structured knowledge into the model's reasoning process without fine-tuning, we adopt a prompting strategy based on in-context learning. Specifically, we append relevant context triples to a base question prompt. This method enables the model to consider factual information from the knowledge graph at inference time. On average, 345 triples from `db-kg` and 75 from `ec-kg` are included in each prompt, with ranges of 18–2,606 and 6–434 triples respectively, depending on the question and retrieved context. On average, 54 triples in the `ec-kg` prompts originated from the text-based triple enrichment step, accounting for approximately 51% of the total triples. The remaining triples were contributed by the structured triple enrichment process.

In our user study, we used GPT-4[17]. In April 2024, when we conducted the user study, GPT-4 was among the top-performing LLMs available[18], with strong capabilities in reasoning, coding, and general language understanding. At that time, GPT-4 was widely recognized for its performance across various benchmarks, making it a suitable choice for our user study.

The participants were asked to assess the quality of the answers with these criteria: (1) granularity of the information, (2) relevance to the question (3)

[16] All scripts are in this folder: `https://github.com/SonyCSLParis/graph_search_framework/tree/main/experiments_run/usage_ng`

[17] `https://openai.com/research/gpt-4`

[18] `https://www.trustbit.tech/en/llm-leaderboard-april-2024`

Table 3.15: Context triples retrieval for DBpedia and the event-centric KGs built by ChronoGrapher. The context triples are retrieved through *CONSTRUCT* queries or equivalent implementation using the HDT data format.

#	Type	DBpedia	ChronoGrapher
1	Summary	Ingoing and outgoing triples of the event	Temporal, spatial, and frame-based contexts
2	Causal explanation	Ingoing and outgoing triples of the event	Frame annotations for causal frames (e.g. `frame:Causation`)
3	Event types within a time range	Events of type `dbo:Event` with timestamps, filtered by time range	Events filtered using start/end timestamps
4	Sub-events	Linked via `dbo:isPartOf MilitaryConflict` and their properties	Events connected with `sem:subEventOf`, enriched with abstracts and temporal data
5	One actor in an event	Events where the actor appears in subject or object position	Events with `sem:hasActor` or frame-mapped role mentions linked to the actor
6	Two actors in an event	Events where both actors co-occur in triples	Events where both actors co-occur as `sem:hasActor`

Table 3.16: Qualitative evaluation metrics used in the user study. Grounding was assessed by us only.

Metric	**Definition**	**Scoring guidelines to the participants**
Granularity	Level of specificity in the answer (generic vs. specific references)	Is the answer generic or specific? (mention of generic events vs. mention of specific events). A specific answer should get a high score, whereas a generic answer should get a low score.
Relevance	Degree to which the answer addresses the question	Do the answer provide an actual answer to the question? If so, should get a high score, else a low score.
Succinctness	Clarity and brevity of the answer	Is the answer brief and clearly expressed? If so, it should get a high score, else a low score.
Diversity	Lexical and semantic variation in the answer	Is the answer varied in content, or very repetitive? It should get a high score if the answer is diverse, and a low score if it is repetitive.
Grounding	Extent to which the answer is supported by the provided context triples (assessed internally)	-

succinctness and (4) diversity in content, on a scale of 1 to 5. We furthermore manually assessed (5) grounding to factual events. A QA system must most importantly give faithful content, making the groundedness metric (5) a priority. The relevance metric measures how well the answer provides a plausible answer to the question, but not its actual truthfulness.

We gathered 10 participants for our user studies. We reached the participants of the user study throughout our research laboratories. The researchers who participated were based in Paris or Amsterdam, and they had a background in computer science and/or computational linguistics. They were not overly familiar with the questions we had them evaluated, and were familiar with LLMs systems and their pitfalls. We designed 2 forms[19] with 6 questions each, and with 3 answers per question corresponding to the three prompt types. We split the 10 participants evenly across these 2 forms. Each participant rated 18 answers (6 question types · 3 prompt types). Each prompt was assessed 60 times for metrics 1-4 ($6 \cdot 10$), and 12 times for metric 5 ($6 \cdot 2$). Given the scale, we solely focused on the French Revolution.

User Study Results

Table 3.17 presents the average scores for each metric and prompt type. The `base` prompt has the lowest groundedness which undermines its reliability, and we found that it was prone to hallucinations, often about actors outside the LLM's implicit knowledge base. This might be linked to its lowest granularity and highest succinctness scores, due to having less content to work with. For the remainder of the analysis, we discard the `base` prompt because of its low groundedness score.

Table 3.17: Average scores for metrics on the human evaluation. The most important metric is groundedness, as it assesses the faithfulness of the answers.

Type Prompt	Groundedness	Granularity	Relevance	Succinctness	Diversity
`base`	1.11 ± 1.47	(3.92 ± 0.96)	(4.18 ± 1.03)	(4.22 ± 0.92)	(3.83 ± 1.03)
`db-kg`	2.24 ± 1.6	$\mathbf{4.15} \pm 0.9$	4.02 ± 1.07	3.35 ± 1.01	$\mathbf{3.97} \pm 1.09$
`ec-kg`	$\mathbf{2.85} \pm 1.86$	4.13 ± 0.96	$\mathbf{4.12} \pm 1.01$	$\mathbf{3.62} \pm 0.99$	3.82 ± 1.03

When comparing the triples-based prompts, `ec-kg` stands out with the highest groundedness (+27%) and succinctness (+8%) scores, indicating that its generated answers are both more faithful and concise. Additionally, `ec-kg` scores slightly higher in relevance (+2.5%) compared to `db-kg`, though `db-kg`

[19]The forms links can be found on the bottom of this README: `https://github.com/SonyCSLParis/graph_search_framework/tree/main/experiments_run`.

achieves a higher diversity (+3.8%) score, likely due to its more varied context triples, and a slightly higher granularity score (+0.5%). Overall, `ec-kg` emerges as the most promising approach, balancing faithfulness, relevance, and conciseness effectively.

This first small user study thus hints that event-centric KGs can help an LLM provide more factual and concise answers to event-centric questions (2.85) than a base prompt (1.11) or a generic KG-based prompt (2.24), while maintaining succinctness.

3.5 Discussion

In this section, we reflect on the design choices, assumptions, and limitations of our approach. We begin by discussing the role of EventKG as the ground truth (Section 3.5.1) with an example. We then consider the methodological scope of our framework and motivate our design decisions, particularly the choice of a rule-based approach over learning-based alternatives (Section 3.5.2). Following this, we explore the generalisability of ChronoGrapher, both to new datasets and to different types of events beyond the historical domain we primarily focus on (Section 3.5.3). Lastly, we reflect on the potential uses of the event-centric KG produced by ChronoGrapher, and the kinds of usage and applications it supports (Section 3.5.4).

3.5.1 EventKG as the Ground Truth

As many parts of our evaluation rely on EventKG as a reference, we conducted a qualitative comparison on a few subevents of the French Revolution between subgraphs extracted by ChronoGrapher and their counterparts in EventKG. Our goal was to better understand the complementarity between the two systems.

Figure 3.6 provides a visual comparison between the outputs of EventKG and ChronoGrapher. In the top part of the figure, we observe that ChronoGrapher does not retrieve `dbr:Flight_to_Varennes` and `dbr:Tennis_Court_Oath` that are present in EventKG. Upon inspection, we found that these events are not explicitly typed as `dbo:Event` in DBpedia, which prevents ChronoGrapher from including them during traversal. Despite these omissions, ChronoGrapher also uncovers relevant sub-events absent from EventKG. For instance, it successfully identifies `dbr:Battle_of_Amsteg` and `dbr:Invasion_of_Guadeloupe_(1794)` as sub-events of the French Revolution, which arguably should be considered part of the historical narrative. In the middle part of Figure 3.6, we show that both EventKG and ChronoGrapher include tempo-

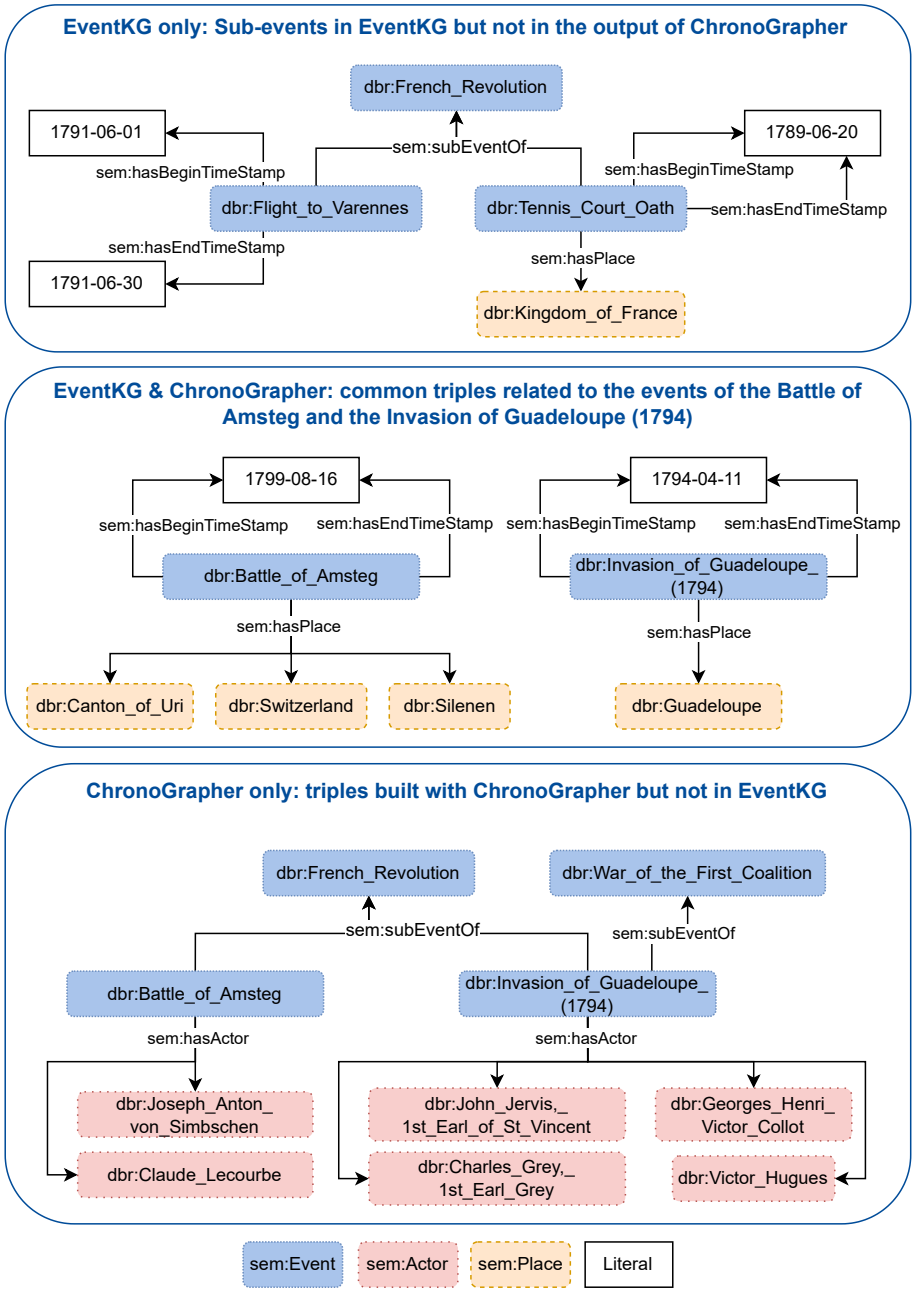

Figure 3.6: Comparison between EventKG and the output of ChronoGrapher with two events.

ral and spatial information. However, as illustrated in the bottom section of the figure, ChronoGrapher provides a more complete representation, offering additional sub-event links and richer actor associations.

3.5.2 Methodological Scope

Our methodology adopts a heuristic, modular approach that prioritizes semantic transparency and temporal control in the construction of event-centric KGs. This design decision is motivated by the requirements of understanding and reasoning tasks, where explainability and domain alignment are essential. While basic in form, the approach performs effectively, as demonstrated in both automatic and user-based evaluations. Moreover, the focus of this work lies not in proposing a novel learning algorithm, but in defining a clear and extensible pipeline for event-centric KG construction and use. By anchoring event representation in a semantically coherent and temporally aware framework, we provide a reliable foundation for downstream learning-based methods, such as embedding models or graph neural networks, that could build upon the structured graphs produced. We believe this balance of simplicity, effectiveness, and extensibility offers practical value for real-world applications while supporting future integration with more complex models.

3.5.3 Generalisibility of ChronoGrapher

We discuss the generalisability of ChronoGrapher, both in terms of its adaptability to different datasets and its applicability to diverse types of events.

Generalisibility to other datasets

ChronoGrapher is flexible and could be applied to other types of nodes or datasets, including those using the PROV vocabulary, which could also yield event-centric graphs. As a reminder, it consists of two components: (1) a pruned, semantically-informed best-first search, and (2) a knowledge graph constructor. The search component is highly configurable through a lightweight configuration file[20] and supports various traversal strategies, such as:

- Semantically-informed or random walks, with or without ground truths;
- Target-based walks in the graph.

[20] An example of configuration file for an HDT dataset can be found here: `https://github.com/SonyCSLParis/graph_search_framework/blob/main/configs-example/config-dbpedia.json`. An example of configuration file for a SPARQL interface can be found here: `https://github.com/SonyCSLParis/graph_search_framework/blob/main/configs-example/config_sparql.json`

This is illustrated in our open-source implementation[21]. The search is also not bound to focus on events only, but can be configured to focus on any type of nodes, such as people or places. Lastly, the search can be run on any KG that is accessible through an HDT dump or a SPARQL endpoint. The constructor is adaptable, provided that predicate labels are retrievable (e.g., via SPARQL).

Furthermore, while ChronoGrapher does not currently implement incremental updates, its design allows for a practical workaround. In dynamic settings (e.g., live sports events), one could extract only the newly added parts of the dataset—centered, for instance, around new event nodes—and rerun ChronoGrapher on these subgraphs. This avoids reconstructing the full event-centric KG from scratch and enables more efficient updates.

Generalisibility to other event types

While our evaluation primarily focuses on historical conflict events, we also explored the applicability of our approach to other domains, notably political and sports events. These types of events often differ structurally from historical ones in KGs: political events (e.g., elections) and sports events (e.g., Olympic competitions) tend to be organized as lists of sub-events (disciplines, rounds, or contests) within a broader event. Moreover, the semantic richness in these domains is often limited, with many relationships captured through generic predicates such as `dbo:wikiPageWikiLink`, which lack fine-grained semantics.

To assess performance in this context, we conducted additional experiments using DBpedia, incorporating `dbo:wikiPageWikiLink` as an accepted relation for traversal. Out of 42 political events and 982 sports events, 5% and 48% respectively received an F_1 score of zero, with average F_1 scores of 56.5% for political events and 22.5% for sports events (considering only one iteration to reduce noise from the `dbo:wikiPageWikiLink` links). These lower performances reveal structural and semantic limitations in the current DBpedia encoding of such events. For instance, we observe that some relevant nodes lack explicit `rdf:type` information. These insights highlight important directions for future work, including better category integration, improved type inference, and more precise handling of weakly-typed links in domain-specific event graphs.

[21]The arguments at the end of the file provide an overview of possible methods for the search: `https://github.com/SonyCSLParis/graph_search_framework/blob/main/src/framework.py`

3.5.4 Usage of the Event-centric KG

In this work, ChronoGrapher was designed to support a wide variety of question types by focusing on events, people, locations, frame-based relationships, and causal interactions. The event-centric KGs can still capture a rich set of relationships between entities and events. This approach allows us to answer questions ranging from straightforward fact retrieval (e.g., Who was involved? or Where did the event take place?) to more complex reasoning questions (e.g., What caused this event? or What were the consequences?).

However, we acknowledge that the types of questions are inherently constrained by the content it contains. As is the case with any knowledge-based task, the quality and scope of the questions answered are determined by the data represented in the knowledge graph. This limitation is a natural consequence of working with structured data, where the knowledge captured reflects the domain's scope and the granularity of its representation. Furthermore, ChronoGrapher's design aligns with the broader goal of narrative construction, which aims to facilitate understanding and reasoning about events within a narrative context. In our previous work, we identified specific requirements for narrative structures that inform the development of such knowledge graphs, ensuring that Chronographer can address a variety of questions within its scope and purpose [31].

Adapting ChronoGrapher to a new dataset requires some familiarity with the target knowledge graph's ontology, particularly to identify predicates analogous to those in the SEM ontology (e.g., for time, location, or actor roles). While this setup currently requires manual configuration, we envision an interface that could assist non-expert users. Such an interface could suggest relevant predicates based on common usage in the dataset and help define event-centric filters through guided forms or templates. In practice, applying ChronoGrapher to new domains may also involve collaboration with a domain curator or a developer familiar with the schema. These additions would help extend ChronoGrapher's applicability beyond users familiar with coding.

3.6 Conclusion

We address the challenge of automatically constructing event-centric KGs from generic ones. We present ChronoGrapher that extracts event-centric subgraphs from generic KGs, and builds event-centric KGs. To extract event-centric subgraphs (RQ1), ChronoGrapher contains a pruned, semantically-informed, best-first search traversal integrating event-centric filters to prune the search space and a ranking step for node prioritisation, yielding F_1 scores of 0.78 and 0.68

for DBpedia and Wikidata. To generate event-centric KGs (RQ2), ChronoGrapher combines a structured triple enrichment based on IRIs and a textual triple enrichments based on abstracts encoded a literals in KGs, achieving F_1 scores of 0.67 and 0.76 for DBpedia and Wikidata, and information extraction from text. To evaluate whether integrating event-centric KGs enhances the performance of LLMs compared to using LLMs alone (RQ3), we conduct first experiments comparing different prompts on an event-centric QA setting, and show that prompts enriched with event-centric KG triples give more factual answers while maintaining succinctness (2.95 vs. 1.11 and 2.24).

Future work will focus on improving ChronoGrapher in a number of directions, including expanding the search method to reach more sub-events using e.g. EventKG for additional type information; expanding the event-centric KG population beyond EventKG to tackle data noise and incompleteness; improving the modelling of complex entities; considering novel ontologies and vocabulary for the event representation (e.g. RDF-star); and use more sophisticated RAG-based methods for a larger user study. We also plan to work on additional quantitative and automatic metrics for a better evaluation of the event-centric KGs.

Supplemental Material Statement: Source code for our system, the baselines we (re-)implemented and the experiments are available on Github[22]. The source code contains the predicate labels that were used to generate the event-centric KG (Section 3.3.2), pointers to download the datasets and more generic statistics on events and their sub-events (Section 3.4.1), the 12 events for parameter selection (Section 3.4.2), additional results on experiments on the end-to-end system combining the event extraction and the graph generation and the manual annotations for the causation frames (Section 3.4.2), and all code, prompts and data linked to the user study (Section 3.4.3).

[22] `https://github.com/SonyCSLParis/graph_search_framework/tree/main`

4

Narrative Complexity in Knowledge Graph Embeddings

Based on [34]:
Inès Blin, Ilaria Tiddi and Annette ten Teije
Measuring the Impact of Narrative Complexity on Knowledge Graph Embeddings
International Semantic Web Conference 2025

In the previous chapter, we introduced a system for building event-centric knowledge graphs to support narrative understanding. In this chapter, we investigate how narrative semantics and graph syntax affect embedding performance. Using a six-level categorization of narrative features derived from Chapter 2, we evaluate different syntactic representations and embedding models, identifying key factors that influence embedding quality.

Abstract. In this work, we study how semantic narrative enrichment and syntactic encoding affect the performance of knowledge graph embedding models. Narratives are central to human understanding, yet their structured representation in knowledge graphs remains challenging due to their semantic and structural complexity. Although traditional knowledge graphs use binary relations, they struggle to capture richer narrative elements, such as *roles* and *causality*. More complex knowledge graph syntaxes such as rdf-star, reification, singleton or n-ary properties offer greater modeling flexibility, but their impact on downstream tasks such as link prediction remains unclear. We define a six-level categorization of narrative semantics and use it to construct a suite of structured knowledge graphs using four syntactic representations. Using different embedding models, we evaluate how semantic and syntactic

factors influence the embedding quality. We find that *semantic* features such as properties and subevents generally enhance performance, while roles tend to have a detrimental effect. On the *syntactic* side, although differences were not statistically significant across all metrics, *reification* and *rdf-star* achieved the strongest results on average.

4.1 Introduction

Narratives are central to human understanding [46]. They serve not only as the main means of communication between humans, but also as cognitive constructs for organizing knowledge and providing coherence to otherwise fragmented data. Narratives thus capture information that goes beyond isolated facts in multiple domains such as legal [110] or historical [238]. Modeling such complex information poses substantial challenges, for instance, with knowledge graphs (KGs). The benefit of representing narratives in a structured format such as KGs lies in their ability to support semantic layering, maintain data lineage, and enable fine-grained reasoning over events, entities, and their interrelations [201].

KGs are widely used to structure factual information through entities and their relations [62, 152], but most KGs rely on binary relations, which limit their ability to capture the semantic depth and internal structure of narratives. Narrative features like *roles* or *causal* links often go beyond what standard formalisms like *rdf* can express [31]. More complex extensions such as *reification* [229], *blank nodes* [76], *singleton properties* [248], and *rdf-star* [145] introduce support for statements about statements. Consider the sentence: "Federer won the match after breaking Nadal's serve in the final set." This narrative encodes not just the *outcome*, but *temporal* order (after), a key *subevent* (breaking serve), and participant *roles* (Federer as winner, Nadal as opponent). Capturing such structure in a KG requires modeling both events and their relations, such as causation and temporal sequence, across multiple semantic layers.

Despite this added expressiveness, modeling narratives with these syntaxes remains difficult: resulting graphs are often large, complex, and hard to query or reason over directly [12, 151]. Embedding methods offer a promising alternative by projecting symbolic structures into continuous vector spaces, facilitating scalable reasoning and prediction [53, 379]. Yet, an open question remains: *to what extent can these embeddings preserve the semantics of rich narrative representations?*

While previous work has addressed narrative representation and construction in KGs [110, 215, 238, 279], there remains limited insight into how varying levels of narrative complexity and the syntactic choices used to encode

them impact downstream tasks such as embedding-based inference. Building on existing work outlining narrative-specific requirements for KGs [31], we model this variation through a six-layer categorization (*base, properties, subevents, text, roles* and *causal relationships*), incrementally adding semantic depth and syntactic richness. This categorization guides the construction of narrative KGs and the evaluation of semantic and syntactic effects on embedding performance. Our investigation centers on two key research questions:

RQ1 (Semantics): How does increasing the semantic richness of narrative representations affect the performance of KG embeddings?

RQ2 (Syntax): How do different KG syntactic constructs, such as *reification* or *rdf-star*, impact embedding performance?

For each of the research questions, we hypothesize the followings:

H1 (Semantics): Increasing semantic richness will improve performance, but gains might diminish as complexity increases, suggesting limits in model capacity to leverage rich narrative features.

H2 (Syntax): More complex constructs like *rdf-star* will outperform simpler encodings in scenarios with complex semantics, due to their ability to natively represent structured relationships. However, this benefit may come at the cost of higher computational overhead.

This paper makes the following contributions:

- We introduce a categorization for modeling narratives in KGs, capturing increasing levels of semantic and syntactic complexity;
- We construct a comprehensive suite of narrative KGs across diverse representation strategies and semantic layers;
- To our knowledge, this is the first empirical study to systematically examine the impact of narrative semantics and KG syntactic constructs on embedding performance, addressing a gap in the existing literature.

4.2 Knowledge Representation Models

We outline several KG representation models that extend *rdf* to support statements about statements. We first recall the standard definition of a KG.

Definition 4.1 (Knowledge Graph)**.** A knowledge graph (KG) is an *rdf* graph where the nodes represent entities (subjects or objects) and literals (objects), and the edges represent the relationships (predicates) between these entities and literals. Formally, $KG = \{(s,p,o)|s \in E, o \in E \cup L, p \in R\}$, where E is the set of entities (nodes) in the KG, L is the set of literals in the KG, and R is the set of predicates (edges) representing relationships in the KG.

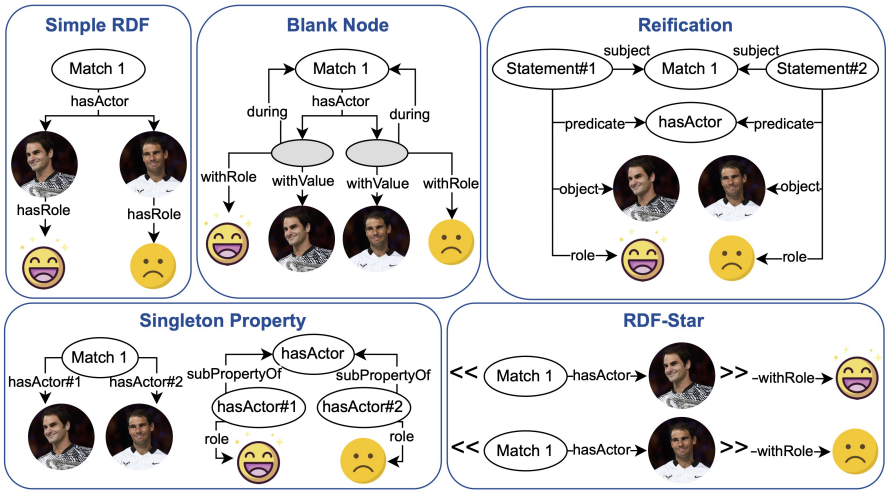

Figure 4.1: Examples of existing syntactic representations.

To illustrate the differences between different representations, we consider a simple scenario. Roger Federer and Rafael Nadal play a match in which Federer wins and assumes the "happy" role, while Nadal loses and takes the "sad" role. Figure 4.1 shows how to encode the statement using five syntactic models: *rdf*, *blank node*, *reification*, *singleton property*, and *rdf-star*. Each model provides a distinct mechanism for attaching additional contextual information.

Simple RDF (*rdf*) [76] represents facts as atomic triples (s,p,o), where a single predicate links a subject and object. It is widely used for its simplicity, but lacks built-in mechanisms for attaching metadata to statements.

Blank Nodes (*bn*) [76] are anonymous resources that group related information without global identifiers. While they offer flexibility for modeling composite structures, they can complicate querying and provenance tracking.

Reification (*reification*) [229] creates a statement node that connects to the subject, predicate, and object of a triple using *rdf*'s built-in vocabulary. This allows metadata attachment but increases verbosity and complexity.

Singleton Property (*sp*) [248] generates uniquely identified predicates (e.g., `hasActor#1`) for each triple, allowing metadata annotation.

RDF-Star (*rdf-star*) [145] extends *rdf* by embedding triples as subjects or objects in other triples, offering a compact way to represent metadata without auxiliary nodes or properties.

Table 4.1 outlines key syntactic properties of each KG representation, including their impact on triples, entities, and properties. We define five qualitative criteria and introduce a scoring system from 1 (low) to 3 (high) to evaluate them. The criteria are: *Modelling* (ease of information encoding), *Queryability* (ease of querying), *Compactness* (graph size efficiency), *Richness* (semantic depth and expressiveness), and *Tooling Support* (availability of practical libraries and tools). Scores are derived from our synthesis of insights in documentation and prior literature [10–12, 80, 156, 161, 169, 253].

Table 4.1: Comparison of KG Syntax Constructs. t: triples, e: entities, and p: properties. *bn* include two additional blank nodes.

Syntax	#t	#e	#p	Modelling	Queryability	Compactness	Richness	Support
rdf	4	5	2	★★★	★★★	★★☆	★☆☆	★★★
bn	8	5	4	★★☆	★☆☆	★★☆	★★★	★★☆
reification	8	8	4	★☆☆	★☆☆	★☆☆	★★★	★★☆
sp	6	8	5	★★☆	★☆☆	★★☆	★★★	★★☆
rdf-star	2	5	2	★★★	★★★	★★★	★★★	★☆☆

4.3 Related Work

We review three key areas relevant to our study: inductive KG embeddings, models designed to capture complex syntactic structures beyond *rdf*, and the impact of KG syntax on embedding performance.

Inductive Knowledge Graph Embeddings. Earlier methods primarily addressed transductive settings, where all entities are seen during training. In contrast, inductive models generalize to unseen entities, making them better suited for narratives with emerging entities and evolving contexts. Therefore, we focus on inductive models in this work and our comparisons. GRAIL [324] is an inductive embedding model that uses graph neural networks to learn relational representations based on entity interactions, enabling generalization to unseen entities but not relations. NBFNet [386] incorporates the Bellman-Ford al-

gorithm for path-based reasoning, A*Net [385] learns a priority function to guide efficient path exploration, RED-GNN [380] applies attention over relational digraphs, and AdaProp [381] introduces adaptive propagation to filter irrelevant entities. More recent work, such as InGram [202] and TRIX [382], extends inductive reasoning to both entities and relations via relation graphs and expressive triple encodings. Universal models like ULTRA [121] and KG-ICL [75] aim for generalization across heterogeneous KGs without retraining, and RMPI [128] enables direct message passing across relations. For a comprehensive taxonomy, see recent survey [210].

Complex KG Embeddings. Recent advances in KG embeddings have moved beyond binary triples to better capture complex semantics [13, 165, 187, 210, 285, 363]. We focus on three relevant directions: *hyper-relational embeddings* support statements about statements, typically to a single level, while *rdf-star* allows recursive nesting which most models do not yet exploit; *hypergraph embeddings* generalize edges to n-ary relations, enabling constructs such as *bn*; and *text-based embeddings* enrich semantic representation by incorporating textual descriptions of nodes and relations. Many of these models are transductive, yet they offer valuable insights for modeling semantically rich, structurally complex narrative graphs. Table 4.2 lists, to the best of our knowledge, all existing models that support extended KG representations falling into one of these three categories. Most target single-entity link prediction, though some diverge: RAE [377] performs multi-entity prediction, DHNE[332], HyperGCN [367], and Hyper-SAGNN [378] address multi-class classification, and BLP [81] also supports entity classification and retrieval. We highlight inductive methods in the table due to space constraints. Text-based inductive models include SimKGC [346], StATIK [231], and BLP [81], which leverage pretrained language models to encode textual features. MetaNIR [353] applies meta-learning to hyper-relational inductive prediction, and [6] explores joint modeling of structure and text in hyper-relational settings. In this work, we draw on these approaches to analyze how different syntactic constructs and text-enriched representations affect embedding performance.

Impact of Syntax on Embedding Performance. While *rdf* representations are widely adopted, their syntactic variants pose challenges and trade-offs [160, 161]. Recent work has begun to assess how *rdf*, *reification*, *n-ary* relations, or *rdf-star* affect downstream tasks like link prediction. [169] compare several syntaxes across tasks such as querying and graph completion, finding that *rdf-star* and Wikidata formats aid usability, while *reification* performs more consistently in embedding tasks despite its verbosity. Other studies [156] focus on Wikidata, noting the complexity of *sp*, IoT KGs [337] or research suggestions [7]. Comparisons between *rdf* and Property Graphs also show *rdf*'s

Table 4.2: Overview of methods handling complex semantics, categorized by modeling paradigm: Translational (distance-based), Tensor (bilinear/multilinear), NN (neural network-based), Hyperbolic (non-Euclidean geometry), Transformer (attention-based), Walk (random-walk sampling), VAE (variational autoencoders), and GloVe+CNN (textual embeddings). HR: hyper-relational, HG: hypergraph. †: paper not available, ‡: code not available. inductive, and nested.

Data	Type	Systems
HR	translational	m-TransH [355] ; RAE‡ [377] ; HSimplE, HypE [106]
HR	tensor	GETD [216] ; S2S‡ [92], RAM [217]
HR	NN	DHNE‡ [332] ; NaLP†‡ [141], HyperGCN [367] ; StarE [120], HINGE [280], NeuInfer‡ [140], G-MPNN [366] ; [373], Ali et al. [6], GRAN [348] ; HAHE [223], ShrinkE [365], HyperFormer [162] ; MetaRH [352], sHINGE‡ [222] ; MetaNIR‡ [353]
HR	hyperbolic	HYPER2‡ [369]
HR	transformer	HyNT [72]
HR	walk	RDFStar2Vec [99]
HG	NN	Hyper-SAGNN‡ [378] ; NHP†‡ [368] ; HyConvE‡ [345], RD-MPNN [384], Att-ImpGCN‡ [347], PosKHG‡ [69] ; HJE‡ [208], NestE [364], RHKH‡ [350], SLR†‡ [60], HC-Net [164]
HG	tensor	LHP‡ [371]
HG	hyperbolic	H2seqRec [207]
HG	VAE	HeteHG-VAE†‡ [104] ; HyperMLN [68]
Text	GloVe + CNN	DKRL [362]
Text	transformer	BLP [81] (2021) ; SimKGC [346], StATIK [231]

advantages in query efficiency [10, 80] and interoperability [11, 12]. [103] survey SPARQL query relaxation techniques across various syntaxes, and [50] examine OWL-based event modeling. [98] directly evaluate *reification*, *sp*, and *rdf-star* for link prediction in hyper-relational KGs. Most relevant to our work are [169] and [98]. The former targets consumer use cases, without narrative

content or support for *rdf-star* or singleton property embeddings. The latter examine formal models for link prediction in hyper-relational KGs, while we focus on semantically layered narrative KGs and compare diverse embedding approaches, including text-augmented methods.

4.4 Narrative Semantics in Knowledge Graphs

Narratives are multi-layered constructs that vary in complexity based on their semantic richness and depth. Building on the narrative requirements from our previous work [31] (see Chapter 2), we propose a six-level categorization to systematically capture these layers, and *examples* related to the Wimbledon Championship (WC):

1. Base: fundamental elements (e.g. events, actors, locations)
   2006 WC (event), Federer (actor), Wimbledon (location)

2. Properties: simple binary relations between base elements
   (WC, location, Wimbledon)

3. Subevents: complex, recursive event structures
   (2006 WC final, subEventOf, 2006 WC)

4. Text: natural language descriptions providing contextual information
   "Federer wins his fourth consecutive title."

5. Roles: higher-level semantics linking objects
   The roles described in Section 4.2

6. Causal relationships: higher-level semantics between events
   "A close prior match pushed them to elevate their games in the next."

We assume all KGs include the "base" layer by default. The remaining five layers can be selectively activated to control the semantic richness of a narrative representation. We formalize this selective activation using a configuration vector η, defined as a tuple of Boolean variables, where each component of η toggles a narrative layer, enabling systematic control over the semantic KG complexity:

$$\eta = (\texttt{prop}, \texttt{subevent}, \texttt{text}, \texttt{role}, \texttt{causation}) \in \{0,1\}^5$$

Levels 1–4 are expressible in standard *rdf*, though level 4's textual literals pose challenges for embedding models. Levels 5 and 6 require higher-order semantics like statement about statement, demanding more expressive syntaxes

such as *sp*, *reification*, *bn*, and *rdf-star*. To capture this distinction, we define the set of valid syntactic representations as a function of the configuration vector η:

$$\sigma(\eta) = \begin{cases} \{\textit{rdf, sp, reification, bn, rdf-star}\} & \text{if } \texttt{role} = 1 \text{ or } \texttt{causation} = 1 \\ \{\textit{rdf}\} & \text{otherwise.} \end{cases}$$

To construct KGs, we take as input a semantic configuration, a corresponding syntactic representation, an event, and DBpedia as the input KG. The process unfolds in two stages. First, an intermediate, *rdf* KG is built based on the configuration and event. This involves extracting relevant components based on our method ChronoGrapher [35] (see Chapter 3): **(1) base** relations are SEM-centric outgoing links; **(2) properties** are outgoing links from the DBpedia ontology; **(3) subevents** are filtered up to two-hop events surrounding the input; **(4) text** comes from `dbo:abstract` and `dbp:result`; **(5) roles** and **(6) causal** outcomes are obtained via frame-based information extraction from textual sources. In the second stage, this intermediate graph is converted into a target KG according to the specified syntactic representation.

Figure 4.2 illustrates an example where the selected narrative layers include **subevents** and **causal outcomes**, in addition to the **base** layer. This corresponds to a configuration vector $\eta = (0,1,0,0,1)$, with syntactic options $\sigma(\eta) = \{\textit{rdf}, \textit{sp}, \textit{reification}, \textit{bn}, \textit{rdf-star}\}$. We choose *rdf-star* as the target syntax and use the *2006 Wimbledon Championship Final* as the input event. On the left, the figure highlights how each narrative layer is sourced from different parts of the background KG. The center shows the intermediate graph, which integrates extracted textual information and introduces auxiliary nodes to capture provenance. On the right, the final output KG combines standard and *rdf-star* triples.

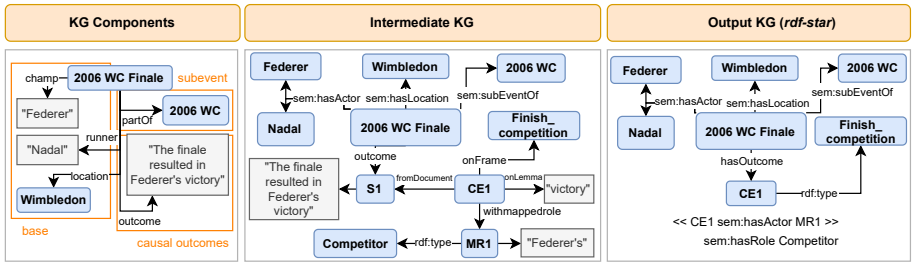

Figure 4.2: Example KG construction for the 2006 WC Finale using the configuration $\eta = (0,1,0,1,1)$ and *rdf-star* syntax.

4.5 Experimental Setup

We present our experimental protocol, covering the embedding methods used (Section 4.5.1), dataset preparation (Section 4.5.2), and training strategy (Section 4.5.3).

4.5.1 Embedding Methods

Given the variety of narrative configurations and syntactic encodings, we focus on three representative, openly available embedding methods, each selected for its compatibility with specific features:

- **ULTRA** [121] is a GNN-based model designed for entity-rich graphs, and supports *rdf*, *reification*, and *sp* syntax.
- **SimKGC** [346] incorporates textual descriptions into KG embeddings, and supports *rdf*, *reification*, and *sp* syntax.
- **ILP** [6][1] is designed for handling hyper-relational forms, and is thus suited for *rdf-star* up to depth 1. It also integrates textual descriptions.

We selected SimKGC due to its recent state-of-the-art performance, ILP as the only publicly available method supporting hyper-relational data, and ULTRA for its recency and general-purpose design across heterogeneous KGs. We exclude *bn* encodings and n-ary relationships from the core experiments due to the absence of open-source inductive embedding methods for these formalisms, which we identify as a valuable direction for future work. We evaluate models using the standard link prediction task: given a triple with one missing element, either the tail $(h, r, ?)$ or the head $(?, r, t)$, the model must predict the correct missing entity by ranking all possible candidates.

4.5.2 Data Preparation

We focus on events classified as *revolutions* in DBpedia [15], as they are complex, multi-layered phenomena involving key actors, causes and consequences, making them suitable for narrative modeling. We extracted 366 entities of type `dbc:yago/Revolution107424109`[2], retaining the 204 (56%) that include both start and end dates. For each revolution and each semantic (see Section 4.4 for the 6 narrative layers) and syntactic (see Section 4.5.1 for the retained models) variant, we build a KG following the method described in Section 4.4, and precompute and store all resulting KG variants.

[1] The system in [6] does not have a specific name, so we use this notation hereafter.

[2] `dbc` is the prefix for *http://dbpedia.org/class/*.

We apply a time-based split: revolutions are sorted chronologically, with 70% assigned to training, 15% to validation, and 15% to test. For instance, the *French Revolution* (late 18th century) is included in the training set, while the *Arab Spring* (early 21st century) appears in the test set. We ensure no triple leakage across splits and restrict the test set to novel facts not seen in training or validation. This setup reflects real-world inference scenarios, especially in historical or policy contexts where future or ongoing events must be interpreted based only on prior knowledge.

For models requiring contextual inference graphs (ULTRA [121], ILP [6]), we enrich the narrative KGs with external ontological triples and isolate all `rdf:type` predicates within the contextual inference graph. This design emphasizes relational and event-level dynamics (e.g., who did what, when, and why) over type-based classification. We further partition validation and test KGs per predicate: 60% for context (inference graph), 40% for prediction (target graph). For models using text-based inputs (SimKGC [346], ILP [6]), we follow [6] and extract entity and relation descriptions (labels and abstracts) from DBpedia. Table 4.3 summarizes the average number of statements produced per configuration. On average, *reification* yields 3.6x more statements than *rdf*, compared to 1.5x for *rdf-star* and 1.0x for *sp*.

4.5.3 Training Strategy and Constraints

We began with pre-experiments using ULTRA and 40 datasets, the only method supporting both zero-shot and fine-tuning setups. Due to the large size of some constructed graphs, we encountered frequent memory issues when processing configurations involving narrative *roles*. A strong correlation (Pearson $\rho = 0.8$, $p << 0.05$) was found between unsuccessful runs and the presence of *roles* in the semantic layer. We conducted our experiments on two Ubuntu machines with comparable configurations, each equipped with 40 CPU cores and 314–376GiB of RAM. The first machine has two GTX 1080Ti GPUs (11GiB), and the second has an A40 (46GiB) and an RTX A6000 (49GiB). Preliminary experiments were run on the second machine. Consequently, we limited the evaluation to configurations without *roles*, resulting in 16 datasets for SimKGC, and 4 for ILP.

For all methods, we adhered to the original parameter and hyperparameter settings from the papers. In cases with excessive parameters, we selected the best-performing models based on the reported results. For SimKGC, we used the *bert-based-uncased* pretrained model[3] as in the original paper [346], and for ILP, we focused on the only inductive model available, StarE [120].

[3] `https://huggingface.co/google-bert/bert-base-uncased`

Table 4.3: Statement counts per semantic layers and syntax. Left columns indicate semantic layers, right-hand cells show statement counts for datasets defined by the combination of those semantic layers and the column's syntax.

prop	*subevent*	*role*	*causation*	*rdf*	*reification*	*sp*	*rdf-star*
0	0	0	0	2,287	-	-	-
0	0	0	1	5,548	8,889	5,925	7,551
0	0	1	0	253,130	615,713	255,082	353,043
0	0	1	1	256,306	622,110	258,259	357,24
0	1	0	0	11,718	-	-	-
0	1	0	1	21,560	43,506	22,177	29,39
0	1	1	0	889,351	3,780,103	891,648	1,450,97
0	1	1	1	899,006	3,811,473	901,304	1,465,40
1	0	0	0	16,949	-	-	-
1	0	0	1	20,210	23,551	20,587	22,21
1	0	1	0	922,681	3,425,791	925,026	1,489,804
1	0	1	1	925,848	3,432,177	928,193	1,493,98
1	1	0	0	67,165	-	-	-
1	1	0	1	77,007	98,953	77,624	84,839
1	1	1	0	2,509,871	20,672,753	2,512,421	5,598,016
1	1	1	1	2,519,516	20,704,111	2,522,066	5,612,425

Table 4.4 summarizes the hyperparameter configurations used for each method. For ULTRA, both zero-shot and fine-tuning experiments were conducted using the existing pretrained variants: *ultra_3g*, *ultra_4g*, and *ultra_50g*. ILP models were trained with embedding dimensions of $\{160, 192, 224\}$, with a maximum of 1000 epochs and early stopping based on validation performance.

4.6 Results

For RQ1 on the impact of narrative semantic richness on embedding performance, we analyze model performance across narrative layers of increasing complexity (Section 4.6.1). For RQ2 on the impact of syntax on embedding performance, we compare results across *rdf*, *reification*, *sp*, and *rdf-star* (Section 4.6.2).

Table 4.4: Hyperparameters used for the three embedding methods: ULTRA [121], SimKGC [346] and ILP [6]. *lr*: learning rate, *bs*: batch size, *bpes*: batches per epoch.

Method	*lr*	*bs*	*epoch*	*bpes*
ULTRA (zero-shot)	-	$\{8\}$	$\{0\}$	-
ULTRA (fine-tune)	-	$\{16,64\}$	$\{1,3,5\}$	$\{100,1000,2000,4000\}$
ULTRA (train)	-	$\{16,64\}$	$\{5,10\}$	$\{1000,2000\}$
SimKGC	$\{1^{-5},3^{-5},5^{-5}\}$	$\{128,256,512,1024\}$	$\{1,10\}$	-
ILP	$\{1^{-3},2^{-3},1^{-4},5^{-4}\}$	{192,256,896,960}	{1000}	-

4.6.1 Impact of Semantic Richness on Embedding Performance

ULTRA. ULTRA is the only method evaluated across all semantic layers, including the most complex narrative elements, such as *roles*. This is possible because ULTRA supports both fine-tuning and zero-shot inference, making it more memory-efficient compared to other models that require full training.

We ran a total of 3,320 ULTRA experiments. Of these, 1,226 runs (37%) completed successfully without exceeding memory limits. Among the successful runs: 106 (8.6%) used zero-shot inference, 1,008 (82.2%) involved fine-tuning, and 112 (9.1%) were trained from scratch. Only 59 of the successful runs included the *roles* layer, all of which used zero-shot inference. We observed a strong positive correlation between the inclusion of properties and performance ($\rho_{\text{MRR}} = 0.78,\ p << 0.05$), suggesting that binary simple relations help structure the narrative and benefit embedding learning. Please note that the properties refer to triples commonly used in large-scale domain embeddings, such as those from DBpedia, which typically yield strong performance. In contrast, more complex elements such as *roles* ($\rho_{\text{MRR}} = -0.30,\ p << 0.05$), *sub-events* ($\rho_{\text{MRR}} = -0.23,\ p << 0.05$), and *causal links* ($\rho_{\text{MRR}} = -0.09,\ p < 0.05$) were negatively correlated with performance, indicating the challenges of embedding semantically dense and structurally complex graphs.

Table 4.5 presents the best evaluation metrics achieved by ULTRA. Each row represents a unique combination of narrative layers, with relative performance changes (δ) compared to the simplest baseline (i.e., base elements only). The table highlights that adding *properties* consistently results in large performance gains, while including *roles* or *causation* often degrades results, especially when combined with other layers.

SimKGC. We ran 8 configurations and 78 experiments in total. *Subevents* showed a consistent and statistically significant positive correlation with all evaluation metrics (e.g., MRR: $\rho = 0.50$, Hits@1: $\rho = 0.48$, all with $p << 0.05$), indicating their positive impact on ranking performance in SimKGC. In

Table 4.5: Best metrics for each dataset run with ULTRA. Left columns (p: `prop`, s: `subevent`, r: `role`, c: `causation`) indicate the semantic layers present. Each row reports the best performance for that dataset across all syntax variants. The δ metrics show the difference compared to the first row.

p	s	r	c	MRR↑	δMRR	H@1↑	δH@1	H@3↑	δH@3	H@10↑	δH@10
0	0	0	0	0.23	0.0	0.2	0.0	0.2	0.0	0.3	0.0
0	0	0	1	0.25	0.02	0.22	0.02	0.24	0.04	0.3	-0.0
0	0	1	0	0.24	0.01	0.22	0.02	0.24	0.04	0.27	-0.03
0	0	1	1	0.24	0.0	0.22	0.02	0.24	0.04	0.27	-0.03
0	1	0	0	0.22	-0.01	0.18	-0.02	0.22	0.02	0.27	-0.03
0	1	0	1	0.24	0.01	0.21	0.01	0.24	0.04	0.28	-0.02
0	1	1	0	0.05	**-0.18**	0.03	**-0.17**	0.05	**-0.15**	0.08	**-0.22**
0	1	1	1	0.05	**-0.18**	0.03	**-0.17**	0.05	**-0.15**	0.08	**-0.22**
1	0	0	0	0.7	**0.47**	0.67	**0.47**	0.73	**0.53**	0.73	**0.43**
1	0	0	1	0.31	0.08	0.28	0.08	0.31	**0.11**	0.36	0.06
1	0	1	0	0.25	0.02	0.24	0.04	0.25	0.05	0.27	-0.03
1	0	1	1	0.25	0.02	0.24	0.04	0.25	0.05	0.27	-0.03
1	1	0	0	0.68	**0.45**	0.65	**0.45**	0.71	**0.51**	0.74	**0.44**
1	1	0	1	0.35	**0.12**	0.3	**0.1**	0.37	**0.17**	0.45	**0.15**
1	1	1	0	0.05	**-0.19**	0.03	**-0.17**	0.05	**-0.15**	0.07	**-0.23**
1	1	1	1	0.05	**-0.19**	0.03	**-0.17**	0.05	**-0.15**	0.07	**-0.23**

contrast, *properties* and *causation* had negligible or inconsistent effects, with correlations near zero and non-significant p-values across metrics.

Table 4.6 shows the best scores per configuration and their improvements over the base KG. Configurations with *subevents*, especially when paired with *properties*, achieved the highest gains. In contrast, including all narrative layers led to performance drops, suggesting that excessive complexity may introduce noise that SimKGC's contrastive learning struggles to manage. Overall, *subevents* proved most beneficial, while *properties* and *causation* had limited impact.

ILP. We focused on configurations involving *causation*, as this narrative layer produces nested statements. While *roles* also introduce nesting, they were excluded based on preliminary experiments. As a result, we only consider configurations with causation and ran 192 experiments in total. Both *properties* and *subevents* contributed meaningfully to model performance. *Properties* correlated strongly with improvements across all metrics, including Hits@1 ($\rho = 0.72$), MRR ($\rho = 0.66$), and AMR ($\rho = -0.64$, all $p < 0.05$). *Subevents* notably improved AMR ($\rho = -0.33$, $p < 0.05$). The Adjusted Mean Rank

Table 4.6: Best metrics for each dataset run with SimKGC. Left columns (`p`: `prop`, `s`: `subevent`, `r`: `role`, `c`: `causation`) indicate the semantic layers present. Each row reports the best performance for that dataset across all syntax variants. The δ metrics show the difference compared to the first row.

p	s	r	c	MRR↑	δMRR	H@1↑	δH@1	H@3↑	δH@3	H@10↑	δH@10
0	0	0	0	0.43	0.0	0.34	0.0	0.47	0.0	0.67	0.0
0	0	0	1	0.46	0.03	0.35	0.01	0.52	0.05	0.67	0.01
0	1	0	0	0.45	0.01	0.32	-0.02	0.51	0.04	0.72	0.06
0	1	0	1	0.42	-0.02	0.31	-0.03	0.47	-0.0	0.63	-0.03
1	0	0	0	0.24	-0.2	0.15	**-0.19**	0.25	**-0.22**	0.43	**-0.24**
1	0	0	1	0.32	**-0.12**	0.25	-0.09	0.35	**-0.12**	0.46	**-0.21**
1	1	0	0	0.34	-0.09	0.26	-0.08	0.37	**-0.1**	0.52	**-0.15**
1	1	0	1	0.33	-0.1	0.24	-0.1	0.36	**-0.11**	0.51	**-0.15**

(AMR) [24] is a recently proposed metric that contextualizes mean rank by comparing it to the expected rank of a random scoring model, that was computed with ILP.

These findings highlight that *properties* and *subevents* help enhancing performance in hyper-relational settings. Table 4.7 presents the top-performing configurations, with the best overall results (MRR = 0.32, Hits@10 = 0.48) achieved when both *properties* and *subevents* were included.

Table 4.7: Best metrics for each dataset run with ILP. Left columns (`p`: `prop`, `s`: `subevent`, `r`: `role`, `c`: `causation`) indicate the semantic layers present. Each row reports the best performance for that dataset for the *rdf-star* syntax.

p	s	r	c	AMR↓	MRR↑	H@1↑	H@3↑	H@10↑
0	0	0	1	0.09	0.23	0.14	0.27	0.4
0	1	0	1	0.07	0.25	0.15	0.29	0.43
1	0	0	1	0.04	0.29	0.19	0.34	0.46
1	1	0	1	0.03	0.32	0.24	0.34	0.48

Lastly, to provide a comparative overview of model behavior across shared configurations, Table 4.8 reports MRR scores for ULTRA, SimKGC, and ILP over the common subset of narrative configurations. It is a standard metric in KG embedding that reflects how highly the correct entity is ranked, providing a robust summary of model performance. The results highlight complementary strengths: ULTRA benefits most from properties (e.g., 0.70 MRR with

properties alone, v.s. 0.24 for SimKGC), while SimKGC performs better with *causation* and *subevent* (e.g., 0.46 MRR with causation alone, v.s 0.25 for ULTRA). ILP, evaluated only on configurations involving *causation*, generally underperforms compared to other models. However, its performance improves when both *properties* and *subevents* are included, reaching a peak MRR of 0.32.

Table 4.8: MRR↑ metrics comparison across the three methods. Left columns (`p`: `prop`, `s`: `subevent`, `r`: `role`, `c`: `causation`) indicate the semantic layers present, right-hand cells show the best metric for each method.

p	s	r	c	ULTRA	SimKGC	ILP
0	0	0	0	0.23	0.43	-
0	0	0	1	0.25	0.46	0.23
0	1	0	0	0.22	0.45	-
0	1	0	1	0.24	0.42	0.25
1	0	0	0	0.7	0.24	-
1	0	0	1	0.31	0.32	0.29
1	1	0	0	0.68	0.34	-
1	1	0	1	0.35	0.33	0.32

To answer RQ1, our findings show that semantic narrative enrichment has a measurable but uneven impact on embedding performance. *Properties* improved results in ULTRA and ILP, but had little effect in SimKGC. *Subevents* boosted performance in text-based models (SimKGC and ILP) but reduced it in ULTRA. *Roles* consistently degraded performance (based on ULTRA) and were excluded from other models for feasibility. *Causation* showed no consistent effect. Aggregating top-performing configurations, we found a significant positive correlation between textual inputs and performance (e.g., MRR: $\rho = 0.45,\ p < 0.05$), indicating that textual enrichment benefits models that can leverage it. Still, overall scores remain modest, underscoring the difficulty of encoding complex narrative semantics in current embedding frameworks.

4.6.2 Impact of Syntax on Embedding Performance

We evaluated how syntactic representation affects embedding quality, focusing on configurations involving *roles* or *causation*, features expressed across multiple syntactic forms. For each method and configuration, we retained only the best-performing variant (based on MRR) to ensure fair comparison.

Table 4.9 reports the average performance across four syntactic strategies. *reification* and *rdf-star* achieved the highest scores across metrics. A Kruskal–Wallis test [196] confirmed a significant effect of syntax on Hits@1 ($\chi^2 = 8.51$, $p = 0.037$). Follow-up pairwise Mann–Whitney U tests [227] revealed that *reification* outperformed both the *sp* and *rdf* syntaxes with significance prior to Holm–Bonferroni correction [354], but these differences did not remain statistically significant after correction. No other comparisons showed significant differences.

Table 4.9: Averaged metrics for the different syntaxes.

syntax	MRR↑	H@1↑	H@3↑	H@10↑
reification	0.28	0.25	0.29	0.34
rdf-star	0.27	0.18	0.31	0.44
rdf	0.18	0.13	0.19	0.27
sp	0.17	0.12	0.18	0.26

To answer RQ2, we find that the choice of syntactic representation does influence embedding performance, particularly for configurations involving *roles* or *causation*. Surprisingly, in terms of intuition, *reification* yielded the highest average performance across half of the metrics, despite its known verbosity and structural complexity. We hypothesize that models may mitigate this syntactic overhead by leveraging contextual patterns and additional features within the expanded structure. *rdf-star* also performed well, suggesting that more structured encodings help models capture complex relationships more effectively. However, the overall differences between syntaxes were moderate, indicating that while syntax plays a role, its impact is likely mediated by model architecture and input configuration.

4.7 Discussion

Our findings offer valuable insights into how narrative complexity and KG syntax affect embedding performance, but also point to several avenues for future work. While we focused on revolutions as a narrative domain, expanding the KG construction pipeline to other event types or external datasets could strengthen the generalizability of the results. Although we adopted hyperparameters from prior work, our KG configurations differ from standard benchmarks, which may introduce biases or favor certain models. The use of *properties* introduces challenges in controlling the information content, and both

roles and *causation* were extracted from the same textual sources, raising potential concerns about data leakage. Moreover, both SimKGC and ILP rely on textual features, which limits their applicability in settings with noisy or absent text. Lastly, uneven KG completeness may affect the quality of semantic layers, particularly for higher-order constructs.

Another consideration is that performance metrics are computed over test sets that vary across configurations due to differences in KG construction. While trends across methods and representations are informative, the metrics do not reflect performance on a fixed set of triples, making it difficult to isolate the impact of semantic or syntactic variations based solely on numerical scores. To address this, future work could explore the use of downstream tasks, such as question-answering with large language models, which may help decouple the effects of semantic and syntactic changes from dataset construction, providing a more flexible evaluation of embedding quality.

4.8 Conclusion

We examined the impact of semantic narrative levels and syntactic representations on three inductive KG embedding methods. We found that certain semantic dimensions, particularly *properties* and *subevents*, can enhance performance, though their effects depend on the model. For example, *subevents* improved results in text-based models but hindered performance in ULTRA, suggesting interactions with textual context. *Roles* consistently reduced performance, while *causation* showed no clear advantages. In terms of syntax, while hyper-relational *rdf-star* offers flexibility, *reification*, despite its verbosity, achieved the best performance. We hypothesize this is due to advanced models compensating with richer contextual encodings.

Supplemental Material Statement: Source code for KG construction and data preparation[4], and extended systems[5] are available on Github. Sample KGs are included in the main repository and all the KGs are available on Zenodo[6].

[4] KG construction and data preparation

[5] ULTRA, SimKGC and ILP

[6] `https://zenodo.org/records/16094768`

5

Social Media Discourse on Inequality

Based on [30]:
Inès Blin, Lise Stork, Laura Spillner and Carlo Santagiustina
OKG: A Knowledge Graph for Fine-grained Understanding of Social Media Discourse on Inequality
International Conference on Knowledge Capture 2023

The previous two chapters focused on the historical use case, where we developed a method for constructing event-centric knowledge graphs and analyzed the role of narrative complexity in embedding models. In this chapter, we turn to the social media domain. We propose an ontology and a knowledge graph tailored for fine-grained understanding of social media discourse, with a specific focus on narratives around inequality. While Chapter 2 identified perspective as a narrative requirement, this thesis focuses primarily on semantic aspects for reasons of scope and feasibility. Among the three use cases, however, this social media setting offers the clearest illustration of how perspectives emerge on top of semantic content.

Abstract. In recent years, social media platforms such as Twitter have allowed people to voice their opinions by engaging in online discussions. The availability of such discussions has garnered interest amongst researchers in analyzing the dynamics on critical topics, such as inequality. Most of the current strategies are, however, limited with respect to conveying the fine-grained opinions of users, focusing on tasks such as sentiment analysis or topic modeling that extract coarse categorizations. In this work, we address this challenge by integrating a Twitter corpus with the output of finer-grained semantic parsing for the analysis of social media discourse. To do so, we first introduce the

OBservatory Integrated Ontology (OBIO) that integrates social media metadata with various types of linguistic knowledge. We then present the *Observatory Knowledge Graph (OKG)*, a knowledge graph in terms of the ontology, populated with tweets on inequality. We lastly provide use cases showing how the knowledge graph can be used as the backbone of a social media observatory, to facilitate a deeper understanding of social media discourse.

5.1 Introduction

In recent years, the proliferation of social media platforms such as Twitter has allowed people to voice their opinions by engaging in online discussions. The accessibility of several of these online resources has piqued the interest of researchers and policymakers. They are now eager to capture and discover perspectives and impactful narratives circulating throughout society on critical topics like migration, war, vaccination, inequality, or climate change. Many works have addressed this interest by publishing dashboards [331], social media datasets [64, 93, 102], and by leveraging automated natural language processing (NLP) strategies for the discovery and analysis of online debates [230, 281, 283, 333].

Traditional NLP strategies for the understanding of online debates have focused commonly on sentiment analysis, opinion mining and topic modeling [256, 323]. Such strategies are limited in their capabilities to convey fine-grained analysis of the opinions of individuals or social groups in the form of narratives, arguments or claims, given that statistical strategies are used to label natural language texts that are vague and imprecise in nature. The complexity of Twitter posts, like the usage of slang and acronyms, further complicates precise interpretation.

Fine-grained text analysis techniques, e.g., named entity recognition [88], relation extraction [383], semantic role labeling [214], frame extraction [301], and dependency parsing [185] can provide fine-grained insights into the specific stances communities take in a debate or narrative. Such techniques allow researchers to retrace the provenance of their findings [358], ensuring the validity of results, reproducibility of experiments, and fostering transparency in research. We argue that the integration of tweet metadata with the output of both fine and coarse-grained NLP analyses into a single integrative network, allows researchers to perform increasingly complex analyses to better understand online debates.

With this goal in mind, this work introduces the *Observatory Knowledge Graph (OKG)*, a knowledge graph (KG) in terms of the *OBservatory Integrated Ontology (OBIO)*, which integrates tweet metadata with various types

of linguistic knowledge and Linked Open Data, such as named entities, dependencies, and PropBank rolesets. We present use cases that demonstrate how the OKG can aid researchers in understanding online discussions on inequalities [387], e.g., perceived causes, driving factors and effects of inequalities, as well as the relations among mentioned entities (people, places, events, organizations, etc.) Thus, the contributions of this paper are twofold:

1. the *OBservatory Integrated Ontology (OBIO)*[1], which integrates tweet metadata with linguistic analysis data.

2. the *Observatory Knowledge Graph (OKG)*, which integrates a Twitter corpus with the output of both fine-grained and coarse-grained NLP analyses, and present relevant use cases. Code to create the OKG is openly available[2]. Due to X's access policies, access to the OKG[3] and the SPARQL endpoint[4] maintained by Triply[5] and the IISG[6], can be provided upon request. Full instructions are available on the Zenodo page.

Such a holistic picture can offer valuable insights into public perceptions, social trends, and the perceived effectiveness of inequality mitigation efforts. It can inform decision-making processes, guide empirical research efforts in the field, and contribute to a more informed debate about inequality.

5.2 Related Work

Discourse on social media. We focus on related work that analyzes discourse in social media, and more particularly on Twitter. First, we present work that is the closest to ours, ie., that uses KGs as intermediate representations. Second, we present work that uses NLP techniques to get insights from social media data, not necessarily in graph format.

TweetsKB [102] and TweetsCOV19 [93] are the work the most similar to ours, the latter being a subset of the former containing Covid-related tweets. They created a RDF(S) model for describing metadata and annotation information for a collection of tweets, and they presented useful use cases such as entity-centric data exploration. The authors of [67] used TweetsKB as a starting point to analyze public attitudes towards controversies, specifically on the

[1] `https://www.w3id.org/okg/obio-ontology/`

[2] `https://github.com/muhai-project/okg_media_discourse`

[3] `https://doi.org/10.5281/zenodo.10034210`

[4] `https://api.druid.datalegend.net/datasets/lisestork/OKG/services/OKG/sparql`

[5] `https://triply.cc`

[6] `https://iisg.amsterdam`

topic of migration, and propose the following components in their pipeline: topic modeling, sentiment analysis, hate/speech detection and entity linking to Wikidata. We extend these models with further fine-grained linguistic information retrieved from tweet texts. More specifically, we add information on semantic roles of entities, as well as sentence grammar.

More recently, a lot of attention has been given to analysing social narratives about Covid19 on Twitter. [64] released a multilingual dataset containing tweets about the coronavirus, and [340] monitored the mood of India during the Covid pandemic starting from tweets. Lastly, [4] analyzed trending hashtags on Twitter with a specific use-case on Covid-19, and mapped the output to knowledge bases like Framester [122].

Tweet2story [51] is an NLP pipeline to automatically extract narratives from tweets in the form of simple graphs. Their pipeline includes: actor entity extraction, time entity extraction, event entity extraction, link extraction and semantic role extraction. Our work is complementary, providing a complete ontology for the KG output. The authors of [224] used the Dutch vaccination debate on Twitter to identify online communities, narratives and interactions. [307] combines an NLP pipeline with network analysis to extract conflicting narrative mechanics from Twitter data.

KG from text. We first present ontologies that were used to model textual data, and more specifically focused on social media data. We then present existing resources.

The NLP Interchange Format ontology [153] was designed to integrate text into KGs, whereas the NLP Annotation Format ontology [118] additionally focused on linking linguistic annotations. The OntoLex ontology [233] models lexical data in the semantic web. To represent social media data in the form of a KG, TweetsKB [102] reused existing ontologies such as the Semantically Interlinked Online Community (SIOC) [45]. The Influence Tracker ontology [273] integrates tweet data with quality metrics about Twitter users. Framester [122] is a frame-based ontological resource that bridges major linguistic resources such as FrameNet and PropBank. In our model and graph, we reuse and extend the existing TweetsKB model, and integrate it with (i) text analysis using [153], (ii) new metrics that are not in the Influence Tracker ontology, and (iii) Framester PropBank rolesets.

In terms of existing resources, [279] proposes an NLP pipeline to build an event-centric KG from news data, using frame semantics. Throughout the BioSampo project, researchers extracted KGs from plenary debates, to analyze parlementary language and culture [299]. TakeFive [3] transforms texts into a frame-oriented KG, and FRED [125] also parses natural text into linked data.

5.3 Knowledge Graph Construction

Here, we describe the OBIO ontology (Section 5.3.1), and the construction and validation of the OKG in terms of the ontology (Section 5.3.2).

5.3.1 The Ontology

The goal of the ontology we build is the inclusion of fine-grained semantics, as a semantic layer on top of the tweet texts. We show five example tweets, use these to describe our ontological requirements which then inform the ontology creation process.

Motivating Examples

We present 5 sentences from tweets in Table 5.1 to show the type of analyses we aim to do with our ontology. Relevant content for the understanding of these sentences can be divided in three categories: **(1) tweet metadata, (2) meaning and (3) grammar**.

Table 5.1: Sentences from tweets on inequality.

Id	Sentence Content
1	*"We see the end of the **trauma** created by our brutal system of race and **gender inequality** "*
2	*"These **unregulated systems** could cause **discrimination** on a massive scale says **Buolamwini**"*
3	*"How're we expected to behave **rationally** in the face of **brutality, inequality,** and **racism** when they get "scared" of a black person reaching for their wallet at a traffic stop."*
4	*"Nothing wrong with wanting to end **inequality**."*
5	*"I'm glad you think we should judge people on **merit**."*

The **tweet metadata** category would include information such as the user who posted the content, information on the user, the date of the tweet, etc. It would also include standards metrics such as the number of likes.

The **meaning** category would typically include entities from sentences, such as *inequality* or *Buolamwini*. In our approach, we aim to go beyond the mere enumeration of entities in a tweet, and to add another semantic layer on top, specifically extracted rolesets: verbs and their arguments. PropBank [188] details provide such fine-grained analyses. PropBank, short for the Proposition

Bank, is a linguistic resource that associates verbs with their arguments, also called semantic roles. An instantiated ensemble with a verb and its filled arguments is called a roleset. For example: sentence 2 from Table 5.1, the verb *"cause"* triggers a roleset, that has *"unregulated systems"* as subject and *"discrimination"* as object. Likewise in sentence 5, *"think"* triggers a roleset that has *"we should judge people on merit"* as object.

Lastly, the **grammar** category represents the grammatical structure of the sentences, more specifically the dependency relationships. Other than facilitating the entity extraction process, the analysis can provide interesting insights like the nesting of PropBank rolesets, as is the case in sentence 4, which triggered two rolesets: *"wanting"* and *"end"*.

Ontological Requirements (ORs)

We derive three high-level ontological requirements from the examples above, directly linked to the three types of analyses that our ontology should enable.

OR1: Model the metadata of the social media ecosystem.

1. Distinguish between a regular tweet, a repost and a reply.
2. Each tweet should have a date.
3. Metrics: sentiment, polarity, subjectivity, number of repost, number of likes.
4. User attributes: account verified, number of followers, number of accounts followed, location.

OR2: Model the meaning of the tweets' content.

1. Link tweets to extracted entities, and link them to external ontologies.
2. Link tweets to PropBank extracted rolesets: triggering verb and its arguments.

OR3: Model the grammar of the tweets' content.

1. Tweet content should be chunked down by sentences and tokens.
2. Dependency relations should be added between tokens.
3. Include: lemma, part-of-speech tag, token index.

Ontology Creation

Following best practices in ontology development [54], we aim to re-use existing models and extend them with classes and object properties. Our core starting point is TweetKB [102], that uses the SIOC [45] ontology. We create new classes, data and object properties if they do not already exist. Apart from TweetsKB, we mainly integrate two other ontologies: NIF [153] for integrating text with KGs and Framester [122] for PropBank-related information. The prefix for the OBIO is `obio`[7].

Ontology Presentation

We show the ontology in Figures 5.1 and 5.2. Figure 5.2 mainly covers OR2-2, while Figure 5.1 covers the rest. The prefixes are given in the Figures. We describe below how we encode the ontological requirements:

OR1-1: `obio:RePost` and `obio:Reply` as new sub-classes of `sioc:Post` for reposts and replies. A repost, or a retweet in the Twitter context, is to share another user's tweets to your followers, while a reply is a direct response to another tweet.

OR1-2: We re-use TweetsKB for this part, with `dc:created`.

OR1-3: We introduce the `obio:post_metrics` data property for metrics related to tweets. We define sub-properties of this property for the following metrics: `obio:sentiment_label`, `obio:polarity_score`, `obio:subjectivity_score`, `obio:nb_repost`, and `obio:nb_like`.

OR1-4: We introduce various data properties to describe the following attributes for a user: `obio:is_verified`, `obio:description`, `obio:location`, `obio:follower`, and `obio:following`.

OR2-1: We re-use TweetsKB with `schema:mentions`.

OR2-2: We integrate text data with Propbank annotations and Framester. We attempt to stay as close as possible to the original Propbank annotations in Framester, with minor modifications[8]. For one annotation of a Propbank roleset, we re-use the `wsj:CorpusEntry` class. The main difference is that instead of using blank nodes to describe mapped roles, we use classes from the NIF ontology, such as `nif:Word` and `nif:Phrase`.

[7] Short for `https://www.w3id.org/okg/obio-ontology/`.

[8] See `http://etna.istc.cnr.it/framesterpage/wsj/wsjpropnetannotations/CE_64700`.

Lastly, similarly to the existing Datatype Property `wsj:onLemma`[9], we create an Object Property `obio:onToken` that goes from a `wsj:CorpusEntry` to a `nif:String`. To enforce a better integration between OR2-1 and OR2-2, we add additional `nif:superString` links, as is shown in Figure 5.2.

OR3-1: We re-use the classes `nif:Word` and `nif:Sentence` from the NIF ontology.

OR3-2: `obio:dependency_relation` is property of `nif:inter` to describe the dependency relations between two words in a sentence. Each dependency relation is then added as a sub-property of `obio:dependency_relation`.

OR3-3: We mostly re-use content from the NIF ontology, and add the data property `obio:hasTokenIndex` to link a token to its index in the original tweet.

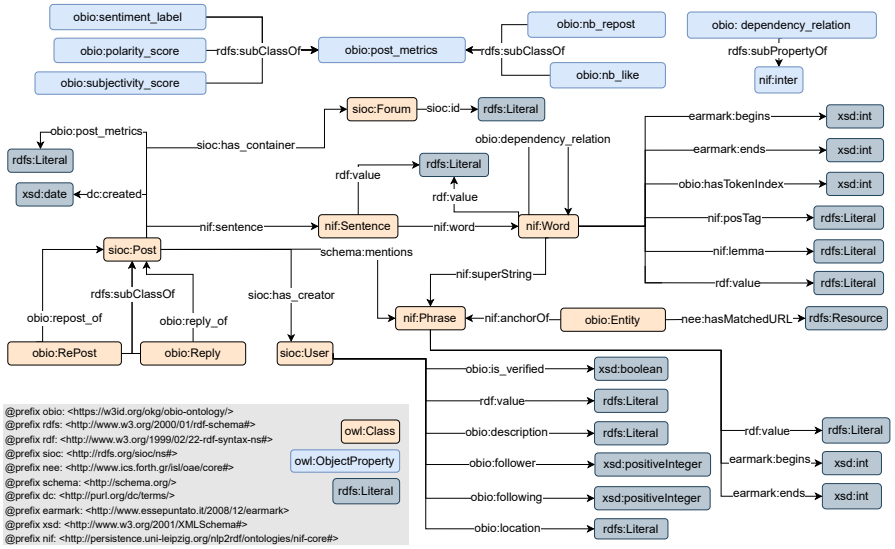

Figure 5.1: The Observatory Integrated Ontology.

Together with the code that we submit with this paper, we add a more detailed documentation and visualization of our ontology.

[9] See `http://etna.istc.cnr.it/framesterpage/wsj/onLemma`.

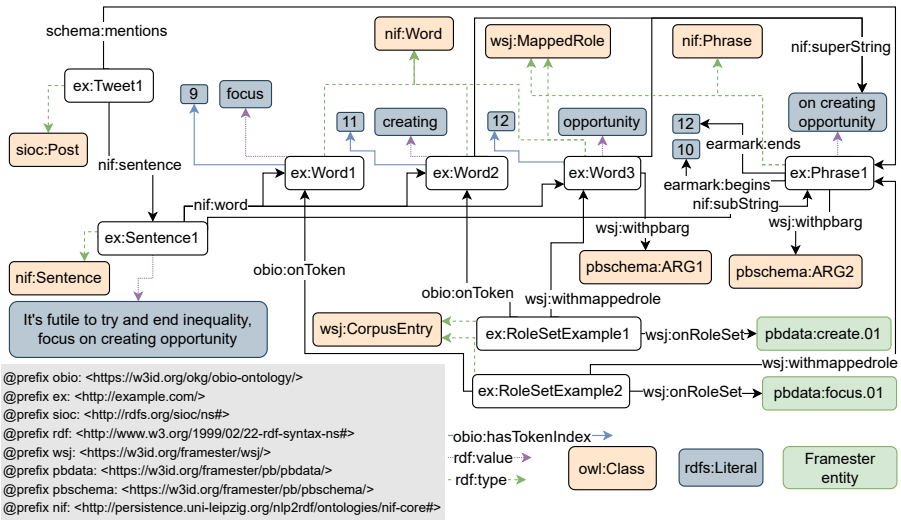

Figure 5.2: Framester integration.

5.3.2 Ontology Population and Validation

Tweet, metadata and grammar extraction (OR1 & OR3).

The acquisition of social media data about inequality from the Twitter platform was done through the "academictwitteR" R library [21], using the Full-Archive API V2[10] with the following query parameters: "*(inequality OR inequalities) lang:en*". We downloaded the data before the changes of policies[11] in June 2023. We retrieved tweets and retweets published from the end (30th) of May 2020 to the beginning (1st) of May 2023. In this paper, we use a sample published from May 30th to August 27th, 2020. To be compliant with the Twitter policies, we remove user metadata and the texts of tweets and tweet sentences. We also replace user IDs with skolem IRIs through skolemization[12].

For the grammar of the tweet content, we used the output of spaCy[13], that covers all requirements for OR3. We used the `en_core_web_sm` model[14]. For the metadata of the tweets, all requirements of OR1 except OR1-3 were provided by the Twitter API: type of tweet (OR1-1), date (OR1-2) and user attributes (OR1-4). For the sentiment extraction (OR1-3), we use the RoBERTa-

[10] `https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/tweet`

[11] `https://developer.twitter.com/en/developer-terms/policy#4-d`

[12] `https://www.w3.org/TR/rdf11-concepts/#section-skolemization`

[13] `https://spacy.io`

[14] Details on accuracy performance can be found at `https://spacy.io/models/en`.

base model trained on around 124M tweets from January 2018 to December 2021 directly from Hugging Face[15], which achieved 69.1% accuracy for sentiment analysis. For the polarity and subjectivity metrics (OR1-3), we use TextBlob[16].

Meaning extraction (OR2).

Entities from text (OR2-1). We use the same methodology as [307]. We first extract two types of named entities from the output of spaCy: named entities and noun phrases. A named entity refers to a proper name, whereas a noun phrase is a grammatical construction that includes a noun and its modifiers. These entity mentions are then consolidated into `obio:Entity` entities, and the mapping to DBpedia is done through DBpedia Spotlight [237].

PropBank Rolesets (OR2-2). To link the tweets to the PropBank rolesets (OR2-2), we use an extended version of the PropBank grammar developed by [27], that uses computational construction grammar to extract semantic frames from text corpora. Such a grammar uses as a basis constructions, that are structured meaning-form pairs. The grammar output the verbs and their semantic roles. We then link the extracted frames to Framester [122]. As an example, the frame `protest.01` from [27] corresponds to the Framester entity `pbdata:protest.01`. For each frame, we create a `wsj:CorpusEntry` as illustrated in Figure 5.2.

Validation and statistics

KG statistics. The KG we present in this paper contains 9,243,293 triples and 1,084,882 unique entities. Out of the 10,613 `obio:Entity` entities, 3,592 (33.8%) had a mapping to DBpedia. The average node indegree and outdegree are 8.4 and 8.5 respectively. The minimum, mean and maximum number of entities per tweets are 1, 2.3 and 11 respectively. 2,398 distinct frames were extracted across all tweets. The top 10 extracted frames were `do.02` (8,029), `pandemic.01` (2,067), `need.01` (1,675), `work.01` (1,674), `see.01` (1,583), `do.01` (1,268), `solve.01` (1,242), `address.02` (1,212), `say.01` (1,179), and `fight.01` (1174).

We describe the distribution of class types in Table 5.2. The most prevalent classes in OKG come from the NIF ontology, which is expected since each tweet was chunked down into sentences and tokens. OKG introduces 136,391

[15] `https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest`
[16] `https://textblob.readthedocs.io/en/dev/`

new corpus entries for PropBank rolesets. There are 62,015 original posts, 34,551 reposts and 4,837 replies for a total number of 42,108 different users.

The KG contains 22,852 tweets with a negative label, 19,070 with a neutral one and 3,927 with a positive one. This represents a ratio of around 6 between the negatively labeled tweets and the positively labeled ones. Furthermore, the average numbers of reposts are 7,015, 827 and 2,769 for the negatively, neutral and positively labeled tweets respectively, which represent a ratio of around 3 between the positive and negative ones. Likewise, the average numbers of like are 2.8, 1.8 and 2.1 respectively, with a ratio of 1.3. We observe the negatively labeled tweets tend to be more numerous, to get more repost and more likes than the other tweets.

Table 5.2: KG Class Distributions.

Class	Number
`nif:String`	787,187
`nif:Word`	573,377
`wsj:MappedRole`	277,502
`wsj:CorpusEntry`	136,391
`nif:Phrase`	107,756
`nif:Sentence`	106,054
`sioc:Concept`	104,123
`sioc:Post`	62,015
`sioc:Forum`	45,472
`sioc:Container`	45,472
`sioc:User`	42,108
`foaf:OnlineAccount`	42,108
`obio:RePost`	34,551
`obio:Entity`	10,613
`obio:Reply`	4,837

Data validation. The resulting graph was validated against a set of data quality criteria [52], specifically *accuracy* and *consistency* given that data was integrated via automated scripts. To check the graph we used SPARQL queries and SHACL shapes. For *accuracy*, we checked the syntactic validity of all literals using a set of SHACL shapes, and a SPARQL ASK query was created to check whether matches to DBpedia were valid URIs, whether all words were indeed included in their superstrings (`nif:superString`), and whether instances of `pbschema:ARG1` and `pbschema:ARG2` were valid Framester IRIs. For *consistency*, we checked for schema correctness using a set of SHACL shapes,

e.g., every tweet has exactly one creator. Errors found through SPARQL and SHACL validation were corrected to improve the quality of the graph. The SPARQL queries and SHACL shapes can be found in the Github repository[17].

5.4 Use Cases

In this section, we present five use cases that reflect examples of questions researchers and policymakers can answer with the OKG. We first present relevant questions that can be answered using sentiment analysis and named entity recognition, similarly to other Twitter resources, e.g., TweetsKB [102]. Second, we present questions solely facilitated by the OKG, through the integration of tweet metadata with finer-grained semantic layers such as PropBank rolesets and dependency relationships. For each use case, we describe their relevance and show the SPARQL queries used to to each use case. For readability we omit prefixes, which can be found in Figures 5.1 and 5.2.

5.4.1 Top Entities Grouped per Tweet Sentiment

SPARQL query 5.1 lists the frequency of sentiment labels per entity mention:

```
SELECT ?entity (COUNT(?t) as ?nb_t) WHERE {
    ?t schema:mentions ?entityMention ;
       obio:sentiment_label ?label .
    ?entity nif:anchorOf ?entityMention .
}
GROUP BY ?entity ?label
ORDER BY DESC(?nb_t)
```

Listing 5.1: Top entities per tweet label.

Table 5.3 shows the top 10 entities mentioned in tweets per label. Some entities are mentioned across all sentiment labels, such as “Economic Inequality”, “Racism”, or “Poverty”. Other entities from the top 10 only appear in one type of tweets, like “Donald Trump” or “Capitalism” for the negatively labeled tweets, or “Coronavirus Disease” for the positive ones.

5.4.2 Entities Co-occurrence Grouped per Tweet Sentiment

SPARQL query 5.2 lists co-occurrence of entity mentions, grouped per sentiment label:

[17] https://github.com/muhai-project/okg_media_discourse

Table 5.3: Top 10 entities in tweets, grouped per label. Freq. corresponds to the frequency of occurrence, and Perc. refers to the percentage of occurrence compared to the total number of tweets.

Negative			Neutral			Positive		
Entity	Freq.	Perc.	Entity	Freq.	Perc.	Entity	Freq.	Perc.
Economic Inequality	609	2.7	Economic Inequality	231	1.2	Racism	75	1.9
Racism	458	2.0	Racism	193	1.0	Economic Inequality	51	1.3
Poverty	234	1.0	Poverty	76	0.4	Poverty	25	0.6
America	117	0.5	Pandemic	63	0.3	Gender Inequality	21	0.5
Institutional Racism	113	0.5	Scientific American Mind	52	0.3	Black Lives Matter	20	0.5
Pandemic	111	0.5	Gender Inequality	51	0.3	Institutional Racism	19	0.5
Donald Trump	99	0.4	Severe Acute Respiratory Syndrome Coronavirus 2	49	0.3	Coronavirus Disease	19	0.5
Gender Inequality	96	0.4	Black Lives Matter	49	0.3	Podcast	14	0.4
Capitalism	94	0.4	Institutional Racism	48	0.3	United Kingdom	13	0.3
Social Inequality	88	0.4	United Kingdom	48	0.3	Web Conferencing	13	0.3

```
SELECT ?sent_label ?ent_1 ?url_1 ?ent_2 ?url_2
       (COUNT(?t) as ?nb_t ) WHERE {
?t schema:mentions ?ent_m_1, ?ent_m_2 ;
   obio:sentiment_label ?sent_label .
?ent_1 nif:anchorOf ?ent_m_1 .
?ent_2 nif:anchorOf ?ent_m_2 .
OPTIONAL {?ent_1 nee:hasMatchedURL ?url_1 .}
OPTIONAL {?ent_2 nee:hasMatchedURL ?url_2 .}
FILTER (STR(?ent_1) < STR(?ent_2))
}
GROUP BY ?sent_label ?ent_1 ?url_1 ?ent_2 ?url_2
ORDER BY DESC(?nb_t)
```

Listing 5.2: Pairs of entities co-occurring.

Table 5.4 shows the top 5 pairs of entities that co-occur in tweets, grouped per sentiment label. Among the negatively labeled tweets, “Economic Inequality” appears in nearly all the pairs, related to other abstract concepts such as “Racism” and “Poverty”. In neutral tweets, entities that appear the most are “Scientific American Mind”, “Happiness” and “Health”. In positive tweets, the set of entities is more diverse, with no clear repetitions.

Table 5.4: Top 5 pair of entities co-occurring in tweets, grouped per sentiment label. Freq. corresponds to the frequency of occurence.

Negative			Neutral			Positive		
Entity 1	Entity 2	Freq.	Entity 1	Entity 2	Freq.	Entity 1	Entity 2	Freq.
Economic Inequality	Racism	42	Scientific American Mind	Terms	45	Coronavirus Disease	Economist	11
Economic Inequality	Poverty	27	Scientific American Mind	Happiness	45	Institutional Racism	Sociology	5
Capitalism	Economic Inequality	23	Scientific American Mind	Health	45	Hurricane Floyd	Minnesota	4
Poverty	Racism	21	Health	Terms	45	130 Crore Indians	British Association For Immediate Care	4
Economic Inequality	Institutional Racism	20	Happiness	Terms	45	British Association For Immediate Care	Modi Govt	4

As presented in Section 5.3, *OBIO* was extended from TweetsKB [102]. OKG can consequently be used for similar use cases than the ones that make use of both metadata information and entities extracted from tweets. For the next uses, we rather focus on the analysis that are enabled by the integration of the PropBank rolesets and the dependency relationships.

5.4.3 PropBank Rolesets per Tweet Sentiment

SPARQL query 5.3 lists rolesets found in tweets, grouped per sentiment label:

```
SELECT ?sent_label ?rs_pb
       (COUNT(?rs_inst) as ?nb_rs) WHERE {
?rs_inst wsj:onRoleSet ?rs_pb ;
         obio:onToken ?lu_token .
?sent nif:word ?lu_token .
?t nif:sentence ?sent ;
   obio:sentiment_label ?sent_label .
}
GROUP BY ?sent_label ?rs_pb
ORDER BY DESC(?nb_rs)
```

Listing 5.3: Number of PropBank rolesets per sentiment label.

Table 5.5 shows the top 10 frames that appear in tweets, grouped per sentiment label. There are differences across the sentiment labels. Whereas the frames for the negative labels relate more to (needed) actions or observations,

such as `solve.01` or `need.01`, the frames for both the neutral and positive labels seem more motivational, such as `support.01`, `fight.01` or `thank.01`. Some frames most frequently appear regardless of the sentiment label, such as `do.02` and `work.01`.

Table 5.5: Top 10 PropBank rolesets from tweets, grouped per sentiment label. Freq.:frequency.

Negative		Neutral		Positive	
Roleset	Freq.	Roleset	Freq.	Roleset	Freq.
`do.02`	7369	`do.02`	1290	`do.02`	582
`pandemic.01`	1690	`need.01`	1002	`work.01`	361
`see.01`	1256	`address.02`	809	`thank.01`	280
`do.01`	1230	`work.01`	777	`fight.01`	278
`solve.01`	1120	`fight.01`	746	`attack.01`	277
`need.01`	1052	`support.01`	699	`admire.01`	276
`work.01`	961	`use.01`	605	`see.01`	257
`expect.01`	933	`change.01`	573	`support.01`	222
`cut.02`	932	`pandemic.01`	570	`love.01`	200
`say.01`	913	`do.01`	569	`help.01`	189

5.4.4 Semantic Roles Linked to Entities

SPARQL query 5.4 lists rolesets linked to entity mentions:

```
SELECT ?rs_pb ?ent ?pbarg
       (COUNT(?rs_inst) as ?nb_rs) WHERE {
VALUES ?link_ss {nif:word nif:subString}
?rs_inst wsj:onRoleSet ?rs_pb ;
         obio:onToken ?lu_token ;
         wsj:withmappedrole ?role_string .
?sent ?link_ss ?role_string .
?t nif:sentence ?sent .
?role_string wsj:withpbarg ?pbarg .

?t schema:mentions ?ent_m .
?role_string nif:superString ?ent_m .
?ent nif:anchorOf ?ent_m .
}
GROUP BY ?rs_pb ?ent ?pbarg
ORDER BY DESC(?nb_rs)
```

Listing 5.4: Linking semantic roles to entities.

Table 5.6 shows the query output and the semantic roles that contain the most entities. "Global Warming" is associated to the roleset `change.01` 112 times in the dataset, with argument `ARG1`. In PropBank, `ARG1` often represents the "Theme" or "Patient" of a predicate, whereas `ARG0` represents the "Agent" or "Experiencer". Most of the entities are strongly related to issues having to do with inequality: various domains of inequality, such as housing or economic inequality, global warming and racial segregation. Lastly, the number of occurrences is not that high compared to the size of OKG, and in particular the size of the corpus entries and the entities. One explanation is that the integration of the entities and the semantic roles can be further refined, as there are cases where entities are substrings of arguments that are not yet linked.

Table 5.6: Top 10 entities included in PropBank rolesets. Freq.: frequency, Neg./Neut./Pos.: negative/neutral/positive.

Roleset	Entity	Pbarg	Freq.	Neg.	Neut.	Pos.
`change.01`	Global Warming	`ARG1`	112	64	38	10
`act.02`	United States Congress	`ARG0`	38	38	0	0
`right.05`	Human Rights	`ARG1`	32	20	8	4
`understand.01`	I Understand 1941 Song	`ARG0`	24	16	6	2
`end.01`	Ibm	`ARG0`	22	8	14	0
`grow.01`	Economic Inequality	`ARG1`	20	18	2	0
`segregate.01`	Racial Segregation	`ARG3`	18	10	8	0
`house.01`	Housing Inequality	`ARG1`	18	8	6	4
`warm.01`	Global Warming	`ARG1`	16	10	6	0
`work.01`	All Facial Recognition Work	`ARG1`	16	6	10	0

5.4.5 Relationships between Rolesets

SPARQL query 5.5 lists pairs of rolesets and their dependency relations:

```
SELECT ?rs_pb_1 ?rs_pb_2 ?dep_prop
       (COUNT(?sent) as ?nb_s) WHERE {
?rs_inst_1 wsj:onRoleSet ?rs_pb_1 ;
           obio:onToken ?lu_token_1 .
?rs_inst_2 wsj:onRoleSet ?rs_pb_2 ;
           obio:onToken ?lu_token_2 .
?sent nif:word ?lu_token_1, ?lu_token_2 .
?dep_prop rdfs:subPropertyOf obio:dependency_relation .
?lu_token_1 ?dep_prop ?lu_token_2 .
FILTER(!CONTAINS(str(?dep_prop), "dependency_relation"))
FILTER(!CONTAINS(str(?dep_prop), "aux")) }
GROUP BY ?rs_pb_1 ?rs_pb_2 ?dep_prop
ORDER BY DESC(?nb_s)
```

Listing 5.5: Dependency relationships between rolesets.

Table 5.7 shows the frames that were appearing the most in tweets with a direct dependency relationship, with the number of occurrences and their distribution across the sentiment labels. Unlike Table 5.6, where most of the content came from the negative tweets, Table 5.7 shows that some rolesets are specifically associated with positive or neutral tweets, despite their lower number. This is the case for `study offers`, `get` and `provide`, and `offering suggestions`.

Table 5.7: Most frequent relationships between rolesets.

Roleset 1	Roleset 2	Dependency	Freq.	Neg.	Neut.	Pos.
`right.05`	`human.02`	`amod`	62	40	16	6
`see.01`	`build.01`	`ccomp`	44	32	12	0
`take.01`	`act.02`	`dobj`	44	22	12	10
`cut.02`	`educate.01`	`compound`	40	40	0	0
`offer.01`	`study.01`	`nsubj`	26	0	4	22
`get.01`	`provide.01`	`conj`	26	0	4	22
`rise.01`	`call.02`	`compound`	26	0	24	2
`help.01`	`move.01`	`ccomp`	26	0	4	22
`offer.01`	`suggest.01`	`dobj`	26	0	4	22
`address.02`	`issue.02`	`dobj`	26	12	12	2

We provided examples of use cases enabled by OKG. Use cases 5.4.1 and 5.4.2 first provided examples of use cases analyzing entities, co-occurrences of entities and sentiment analysis, outlining which actors/objects appear the most in tweets, and are likely to play an important roles in the tweets' narratives.

We then provided use cases to gain a deeper understanding of opinions about entities using finer-grained semantic layers like PropBank rolesets and dependency relationships. In use case 5.4.3, we analyzed commonly used rolesets in tweets, revealing that neutral and positive tweets often contain motivational verbs. In use case 5.4.4, we explored the entities frequently appearing in PropBank rolesets, uncovering connections to various forms of inequality, such as housing inequality and global warming. In use case 5.4.5, we examined the most frequent dependency relationships between PropBank rolesets, discovering links between specific rolesets in positive tweets, despite their low proportion. These use cases demonstrate how integrating tweets with finer-grained parsing allows us to analyze the roles of specific entities in events. Similar insights can be obtained for real-world use cases such as the understanding of online debates on COVID-19 vaccination [66].

5.5 Conclusion

We first present the *OBservatory Integrated Ontology (OBIO)* that integrates social media metadata with various types of linguistic knowledge such as entities and PropBank rolesets. We then populate this ontology with the *Observatory Knowledge Graph (OKG)*, with tweets extracted on the topic of inequality. We lastly present several use cases that show how adding finer-grained semantic layer can help improve the overall understanding on social media discourse. The paper focuses on the topic of inequality but the method is generic and can be applied to other topics.

The work we present is based on a small sample of tweets to emphasise the usefulness of semantic layers. We plan to release a larger-scale KG, improve the integration between entities and PropBank rolesets, and refine representations currently stored in text form through the NIF ontology. Furthermore, our analyzes are based on automatically extracted entities, predicates, and sentiment labels, which can introduce errors and biases. These components are imperfect, and in particular, sentiment and polarity errors can introduce inconsistencies. We aim at further curating the output of the natural language processing, as well as to evaluate the utility of the OKG in a real-world use case with expert users, such as social scientists.

6

Automated Hypothesis Generation

Based on [32]:
Inès Blin, Ilaria Tiddi, Giuliana Spadaro and Annette ten Teije
Automated Hypothesis Generation for Human Cooperation Studies
Hybrid Human Artificial Intelligence 2025

In the previous chapter, we introduced the social media use case, presenting a knowledge graph and an ontology for finer-grained understanding of social media discourse on inequality. In this chapter, we turn to the last use case in the social sciences. We explore how structured representations and knowledge-enriched AI methods can support automated hypothesis generation, helping researchers uncover insights in studies of human cooperation.

Abstract. A meta-analysis is a systematic review of the literature that employs statistical techniques to combine and compare the results of multiple related studies, that can be used to test a hypothesis. In the social sciences, it is the gold standard for research synthesis. However, it requires considerable effort including hypothesis formulation and background research. In our work, we envision a human-AI collaboration process, where AI-generated research hypotheses are integrated in tools that empower domain experts while keeping them in control, fostering trust and efficiency in social science research. In this work, we propose using knowledge-enriched AI methods as an alternative to hypothesis generation to be tested by the meta-analysis, on the specific case study of human cooperation. We collaborated with domain experts to formulate three template hypotheses of increasing complexity. We then compared three knowledge-enriched AI methods, classification, link prediction, and large language model, for hypothesis generation. To evaluate the quality

of the hypotheses generated, we conducted user studies with domain experts, comparing the hypotheses they generated with those generated by AI methods. We demonstrate that AI-generated hypotheses, though not always surpassing human-generated ones, provide diverse, efficiently produced options that effectively complement and enhance the hypothesis generation process.

6.1 Introduction

Recent advancements in AI offer promising avenues to enhance social sciences through new tools and methods, potentially accelerating research processes [257]. Let us imagine a social scientist investigating the impact of economic inequality on cooperative behaviors that benefit the collective, such as paying taxes. Social scientists typically formulate a hypothesis and conduct a meta-analysis that is the gold standard for synthesizing research to test this hypothesis [131]. However, it requires significant effort and is largely a manual process that requires a deep understanding of expert knowledge and statistical and methodological details. Traditional steps of a meta-analysis include: formulation of a hypothesis, search of the literature, selection of studies, choice of dependent variables, selection of a meta-analysis model, data analysis and interpretation of the results [47, 305, 374].

AI-enhanced scientific assistants are gaining traction in research, with support in steps such as analyzing the literature [77, 176, 351], formulating a research question or hypothesis [17, 138, 211, 297], or scientific writing [295]. Large language models (LLMs) are increasingly being used as such assistants [71, 85]. However, despite storing factual knowledge [265], LLMs are prone to hallucinations [163], pushing research to combine symbolic and neurosymbolic methods to address these issues. Recent work [90, 306, 312] has contributed towards a knowledge graph (KG) paradigm based on a structured, interlinked and formal description of the content of research publications [282], which would alleviate background research work. Other works have contributed to automated hypothesis discovery through classification [181] or link prediction tasks [83]. However, current methods lack focus on the social sciences and close collaboration with experts for hypothesis generation, limiting their utility in helping social scientists with their daily tasks.

Figure 6.1 illustrates our vision for human-AI collaboration in hypothesis and meta-review generation. We propose a digital assistant that integrates AI-generated hypotheses, facilitates hypothesis selection, and incorporates user feedback. This design keeps experts in control while handling technical complexity in the background. By streamlining hypothesis formulation, such an AI-powered assistant could enhance social science research and extend to areas

such as policy analysis. Our human-centric approach prioritizes trust, ensuring that AI supports rather than dictates decisions, and actively involves experts in building knowledge-enriched AI methods.

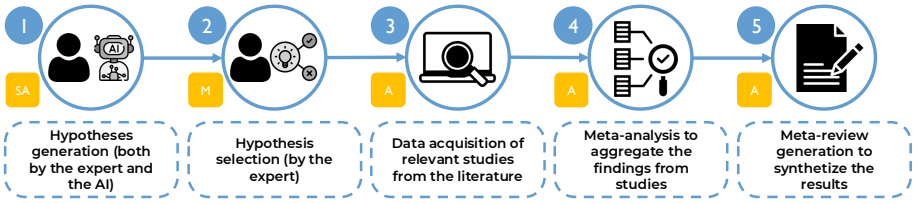

Figure 6.1: Overview of the human-centric process to generate meta-reviews from formulated hypotheses. SA: semi-automated ; M: manual ; A: automated.

This paper focuses on step 1 of Figure 6.1, proposing knowledge-enriched AI methods in collaboration with domain experts. We tackle the complexity of automated hypothesis generation, helping social scientists summarize and aggregate empirical evidence. Our goal is to assess whether AI-generated hypotheses can effectively complement human-generated ones. We define three research questions:

RQ1: What are the requirements for formulating hypotheses?

RQ2: How well do various knowledge-enriched AI methods perfom on hypothesis generation?

RQ3: How well do AI-generated hypotheses compare to hypotheses generated by experts?

Our specific case study aims to understand "human cooperation", a critical aspect of social behavior with extensive implications for economic activities, social policies, and community development [41, 154, 304].

To address **RQ1**, we collaborated with experts to define ontological knowledge constraints and design three template hypotheses. For **RQ2**, we approached automatic hypothesis generation through three distinct methods: classification, link prediction, and LLM prompting, comparing their performance. For **RQ3**, we conducted user studies in which experts generated hypotheses and evaluated and compared them with AI-generated hypotheses. We find that AI-generated hypotheses, though not always surpassing human-generated ones, provide diverse, efficiently produced options that effectively complement and enhance the hypothesis generation process. Our main contributions include a framework for hypothesis generation that combines expert knowledge and a comparative analysis of three knowledge-enriched AI methods. Al-

though we do not introduce new machine learning models, we provide empirical insights into their application and effectiveness in complementing human-generated hypotheses. We show that AI-generated hypotheses can effectively complement those generated by humans. We make our code, experiments, and results openly available[1].

6.2 Related Work

We use *structured scientific knowledge* and contribute to *computer-aided discovery*.

Representing scientific knowledge. The growing volume of publications creates challenges like non-machine-readable data and limited content integration, prompting efforts to structure and interlink research content of publications [282].

We focus on *content-based resource KGs* that encode the content of the articles along with metadata information. ORKG [16] acquires, curates and publishes semantic scholarly knowledge of research papers. CS-KG [90], previously AI-KG [91], is a large-scale, automatically constructed KG of research entities and claims in computer science, and CoDa describes the results of studies in the social science domain [306]. Other work automatically builds resources, such as MIRA [314] on health inequality, Scicero [89] on scientific knowledge, or more generally from unstructured data [199, 226].

Work on *how to represent hypotheses or scientific knowledge* has been done in structured format. Nanopublications are scientific statements with context [137] that are published as RDF reusable data and that are used in domains such as biodiversity [261], physician suicide claims [205] or experimental protocols [130]. The SupperPattern ontology [48] allows researchers to formulate hypotheses as logic statements, while the Neuro-DISK [126] ontology encodes the hypothesis statement, qualifier, evidence, and history.

Computer-aided discovery. Here intelligent systems are used to help scientists derive insights from large datasets, boosting research efficiency and scalability [257]. [189] envisions a Nobel Turing Challenge for an AI system "worthy of a Nobel Prize".

Various *ML methods* have been employed for scientific discovery. Augmented literature reviews, such as ARROWSMITH [318], have been used to stimulate scientific discovery by suggesting complementary articles relevant to

[1] `https://github.com/InesBlin/coda_hypothesis_generation`

a query. We refer to [37] for a review on literature reviews. In the biomedical domain, [18] used various ML models such as decision trees [219] to predict the carcinogenicity of chemical compounds. [308] uses text mining algorithms to discover interesting hypotheses, while [181] frames link discovery as a classification task with learned concepts. [257] proposes a synoptic model to test whether a model and its hypotheses adequately describe reality, and Neuro-DISK [126] is an automated discovery framework based on metaregression.

Hypothesis discovery has also been framed as *graph-based approaches*. [245] used graph diffusion to generate hypotheses from scientific publications. Coco [313] proposes new research hypotheses via a multihop reasoning approach based on reinforcement learning. Link prediction has been used for hypothesis generation [39, 83, 218, 302], leveraging KG embeddings such as TransE [38], DistMult [370], ComplEx [330], and R-GCN [291], or symbolic systems such as AnyBURL [236] that mine first-order rules.

Lastly, *LLMs* have been used as AI scientific assistants for computer-aided discovery. The sole use of LLMs includes domains such as human genomic variations [85], joint arthroplasty [71] or literature reviews [357]. Other systems combine data-driven methods with domain knowledge: [57] extends generative design methodologies to the manufacturing domain, [74] integrates a causal reasoner to guide LLMs in the epidemiological domain, [327] combines causal KGs and LLMs in psychology, and [2] combines data-driven machine learning and KGs for accelerated materials discovery.

We combine data-driven methods with domain knowledge in the social sciences. The closest work to ours [83] frames hypothesis discovery as a link prediction task on knowledge graphs and evaluates hypotheses through user studies in the social science domain. We extend this by categorizing hypothesis discovery into three tasks inspired by the literature: classification, link prediction, and prompting, and consider three types of hypotheses of increasing complexity. To the best of our knowledge, we are the first to cross-study several methods and hypotheses in this domain.

6.3 Designing Meaningful Hypotheses

To answer RQ1 on the requirements to formulate hypotheses, we designed them with experts. We focus on human cooperation and use the **COoperation DAtabank (CoDa)** [306], an open source KG of approximately 3,000 empirical studies from the social and behavioral sciences, each annotated along 312 variables covering study, sample, and effect-size information. CoDa supports efficient search and synthesis of relevant studies, a process that is overwise time-consuming. Table 6.1 defines the CoDa terms reused in this work.

Table 6.1: Definition of terms used by social scientists in their work. Abbr.: abbreviation.

Name	Abbr.	Definition
Effect Size	*es*	A measure that quantifies the difference or relationship between variables in a study.
Generic Independent Variable	*giv*	A category of factors that can be categorized to observe their effect on a dependent variable in various research contexts.
Specific Independent Variable	*siv*	A well-defined variable within a study or context that can be categorized to determine its effect on a dependent variable. In CoDa, *sivs* are instances of *givs*.
Specific Independent Variable Value	*sivv*	The data point or measurement assigned to the *siv*.
Dependent Variable	*dv*	The outcome or response that is measured in an experiment or study to assess the effect of changes in the independent variable(s).

To **design research hypotheses**, we consulted domain experts, who highlighted the importance of comparing the same or related *givs*. Their insights were translated into ontological constraints, as shown in Figure 6.2. The main constraint is that the *sivvs* to be compared are instances of the same *siv*, which is encoded in the ontology.

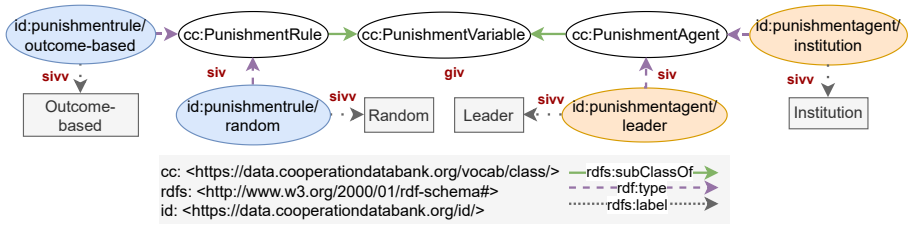

Figure 6.2: One example of ontological constraints for the hypotheses. Only nodes of the same color that are instances of the same *siv* can be compared.

We last developed **three templated hypotheses validated by experts** to confirm their relevance. Figure 6.3 presents the three template hypotheses, `reg`, `study-mod` and `var-mod`, along with examples instantiated with CoDa, such as: "*Cooperation is significantly higher when anonymity manipulation is low compared to when anonymity manipulation is high*". The hypotheses were derived from comparisons of treatments in various CoDa studies.

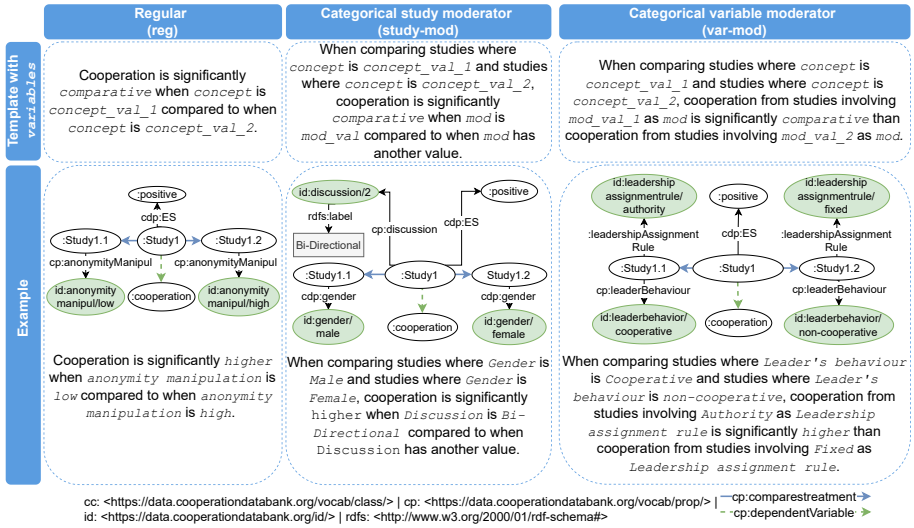

Figure 6.3: Presentation of the three templated hypotheses, where the pairs to be compared must be *sivvs* of the same *siv*, and study moderators are characteristics of study taken from CoDa. The `(:Study1, cdp:ES, :positive)` indicates that cooperation is `higher` (`:negative` would be `lower` cooperation).

6.4 Hypothesis Generation Methods

We outline the data preparation phase (Section 6.4.1) and the methods used, including a classification system inspired by [18] and [181] (Section 6.4.2), a link prediction system from graph-based methods inspired by [83] (Section 6.4.3), and a system combining LLMs for zero-shot prompting [191] with ontological constraints (Section 6.4.4), addressing RQ2 on the effectiveness of knowledge-enriched AI methods for hypothesis generation. The `experiments` folder in our code contains a `README` with all the details to reproduce the experiments, as well as thorough readable details on the results of the experiments.

We selected three families of methods for their diversity and suitability, forming a foundation from which hybrid or more complex systems could be developed. To ensure a manageable scope for subsequent user studies, we prioritized a balance between rigorous method comparison and practical applicability. For each family of methods, we further detail in the sub-sections the different models we compared. The metrics described below (e.g. Hits@10 or MRR) are used solely for internal model evaluation and would not be presented to users in the framework shown in Figure 6.1. Higher metric values indicate a greater likelihood that the hypothesis generated agrees with the existing literature.

6.4.1 Preparing Data

We define criteria similar to [83] to extract information from CoDa for our methods, since we build the hypotheses on top of the existing data. We only kept the observations that (i) reported Cohen's *d* effect size measure, (ii) compared two treatments, and (iii) were related to *dv*. We integrated ontological constraints to extract the data and only considered categorical variables. The `SPARQL` queries used to obtain the observations resulted in 4,463, 91,156 and 846 observations for the `reg`, `study-mod` and `var-mod` hypotheses, respectively. We categorize effect sizes estimates into 5 categories [83].

Each observation is assigned a correlation direction: *positive*, *none*, or *negative*, based on its confidence interval. If the interval includes 0 (nonsignificant), the correlation is classified as *none*; otherwise, it is classified as *positive* or *negative*. For observations without a confidence interval, we used Sawilowsky's rule of thumb [286] to estimate it. Following [83], only the correlations *positive* and *negative* were used in the experiments.

6.4.2 Classification

Classification Task. We learn mappings from hypotheses to their effects on the *dv*. Formally, we define f_{reg}, f_{study_mod} and f_{var_mod} for each of the three hypotheses respectively, where the inputs are concatenations of node embeddings from CoDa of size n. As per the expert constraints described in Section 6.3, $siv_1 = siv_2$. As in Figure 6.3, `positive` and `negative` relate to `higher` and `lower` cooperation respectively.

$$\begin{aligned} f_{\text{reg}} \colon \mathbb{R}^{n\times 5} &\longrightarrow \{\text{positive}, \text{negative}\} \\ (giv, siv_1, sivv_1, siv_2, sivv_2) &\longmapsto y \end{aligned} \tag{6.1}$$

$$\begin{aligned} f_{\text{study\_mod}} \colon \mathbb{R}^{n\times 7} &\longrightarrow \{\text{positive}, \text{negative}\} \\ (giv, siv_1, sivv_1, siv_2, sivv_2, mod, mod\_val) &\longmapsto y \end{aligned} \tag{6.2}$$

$$\begin{aligned} f_{\text{var\_mod}} \colon \mathbb{R}^{n\times 8} &\longrightarrow \{\text{positive}, \text{negative}\} \\ (giv, siv_1, sivv_1, siv_2, sivv_2, mod, mod_1, mod_2) &\longmapsto y \end{aligned} \tag{6.3}$$

Node Embeddings. We trained KG embeddings for the features of the nodes from the ontology. We optimized KG embedding methods using two phases: (1) random search across a set of 500 parameters. (2) Full search on a subset of

these parameters. The **best performance** was with an embedding dimension of 336, a learning rate of 0.001, a negative sample of 1, 500 epochs and DistMult as model. We then saved our training, validation, and test data as embeddings. Appendix A.1.1 provides more details on hyperparameter optimization and optimal values identified for the embeddings.

Model. To ensure a balanced distribution of *givs*, we implement a custom training/validation/test set split. We compared a decision tree classifier, drawing from models used in the literature [18, 181], with more complex models, including multilayer perceptron, k-nearest neighbors, linear models and ensembles. The decision tree achieves the highest validation accuracy for the `reg` (0.78) and the `study-mod` (0.80) hypotheses. For `var-mod`, it scores slightly below the highest score from the k-nearest neighbors (0.70 vs. 0.77, +10%), that scored lower on the `reg` (0.72, -8%) and `study-mod` (0.75, -6%) hypotheses. Given these small differences and the simplicity of the decision tree, we selected it as our final model. Table 6.2 shows the final results, including data size and standard accuracy in the classification task. Appendix A.1.1 provides more details on hyperparameter search, best parameters and results for the complex models.

Table 6.2: Results on the classification task, with a train/val/test split of 0.8/0.1/0.1. Val.: validation.

Hypothesis	`reg`			`study-mod`			`var-mod`		
Metrics	Train	Val.	Test	Train	Val.	Test	Train	Val.	Test
Data size	1,832	241	265	28,235	3,528	3,559	325	43	45
Accuracy	0.79	0.78	0.73	0.80	0.80	0.78	0.74	0.70	0.62

The classification task achieves the highest accuracy score, 0.78, for the `study-mod` hypothesis, which can be accounted for the highest size of training data. In contrast, the lowest accuracy is for the `var-mod` hypothesis: 0.62, slightly higher than the random 0.5. This could be explained by the much lower data size, both for training and testing.

Feature importance measures provide information on the variables that drive our models. Figure 6.4 shows the importance of the grouped features by variable. For the `reg` and `study-mod` hypotheses, we observe similar trends. The main contributors are $sivv_1$ and $sivv_2$, with feature importance scores of 0.32 and 0.20 for `reg`, and 0.21 and 0.39 for `study-mod`. *giv* has a categorization role, but its lower importance highlights that its subclasses (*sivs*) and instances (*sivvs*) are more critical for prediction. A similar pattern emerges

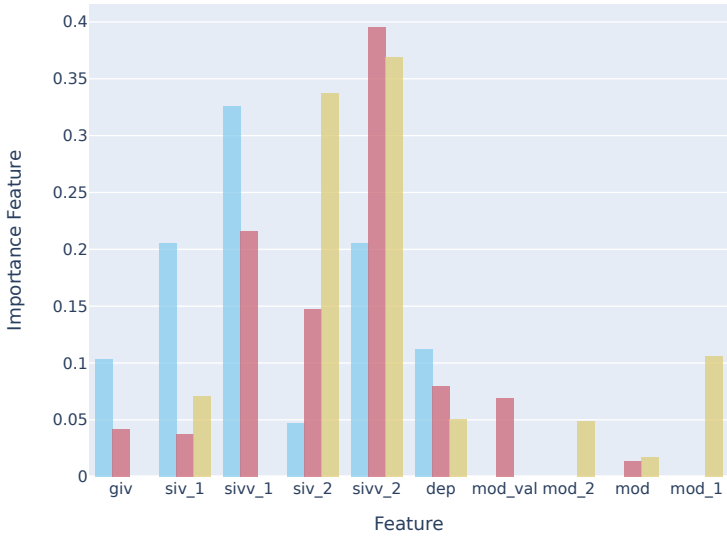

Figure 6.4: Grouped feature importance (y-axis) per variable (x-axis). The `reg` has the first six variables, `var-mod` the first 8 and `study-mod` the first six and the last two. Blue: `reg`, red: `study-mod`, yellow: `var-mod`.

with *mod_val* and *mod*. The lower importance of these moderator variables for `study-mod` reflects the nature of the data: studies often include many moderators but only a few treatment comparisons, leading to repeated effect sizes that amplify the relevance of *sivs* and *sivvs*.

For the `var-mod` hypothesis, the strongest contributors are $sivv_2$ (0.36) and siv_2 (0.34), followed by mod_1 (0.10), which emphasizes the critical role of *sivs* and their instances. However, the lack of influence of $sivv_1$ and *giv* is unexpected. Moreover, the lower overall performance of this model suggests that these features alone are insufficient to capture the complexities of the `var-mod` hypothesis.

6.4.3 Link Prediction

We generate blank hypothesis nodes from CoDa, linked to its *dv* via predicates `cdp:has{Positive,Negative}EffectOn` . To balance the distribution of *givs*, these triples are divided into training/validation/test sets, with the rest in the training set.

We compare (a) symbolic and (b) sub-symbolic models. AnyBURL demonstrates strong performance without further optimization [236], as shown in Table 6.3. For the link prediction task, Hits@10 scores of 0.53, 0.51, and 0.61 and MRR scores of 0.45, 0.44, and 0.54 were achieved for the `reg`, `study-mod`, and `var-mod` hypotheses, respectively. Rules involving *effects*

Table 6.3: Results for the link prediction task, with a train/val/test split of 0.8/0.1/0.1. Hits@k: proportion of times the correct answer is within the top k predictions; MRR (Mean Reciprocal Rank): mean of the inverse of the correct answers' ranks. There are 7,252 triples from the ontology derived from it to describe the blank nodes (only included in the training data). E-1, E-3: Effect rules of length 1 and 3.

Hypothesis	**Data size**			**Metrics**				**Rules learned**			
	Train	Val	Test	Hits@1	Hits@5	Hits@10	MRR	All	Effect	E-1	E-3
`reg`	15,267	301	281	0.3759	0.5124	0.5337	0.4485	2,482	1,095	1,055	40
`study-mod`	284,399	6,223	5,761	0.3828	0.5082	0.5103	0.4449	10,673	2,088	2,040	48
`var-mod`	7,632	53	45	0.5109	0.5326	0.6087	0.5420	2,517	659	499	160

(`cp:hasNegativeEffectOn` or `cp:hasPositiveEffectOn`) account for only 44%, 20% and 26% of the rules for the three hypotheses. This suggests that AnyBURL primarily learns rules about the ontology or hypothesis templates rather than directly modeling their effects on cooperation. Furthermore, most rules are simple (length 1), indicating that they may lack sufficient complexity to be informative. Only in the `var-mod` hypotheses do more complex rules (length 3) appear with notable frequency (32%), compared to just 3.7% and 2.3% for `reg` and `study-mod`.

The support scores for the effect rules are low in all hypotheses (0.14, 0.17, and 0.20 for the `reg`, `study-mod`, and `var-mod` hypotheses), making it challenging to generalize the findings. $sivv_1$ and $sivv_2$ emerge as the most influential contributors, with a significant drop-off before the moderator variables (mod, mod_1, and mod_2). These patterns highlight the potential of AnyBURL to capture some relevant insights, but also underscore its limitations in fully representing the effects hypothesized in social science research. Appendix A.1.2 provides more details on support scores, metrics and rules learned by AnyBURL.

For the subsymbolic models (b), we applied the same methodology as for the KG embeddings and first applied a random search on 100 parameters for TransE [38], DistMult [370], ComplEx [330], and R-GCN [291]. We discarded these models because of their significantly lower performance compared to AnyBURL. The best MRR scores achieved were 0.01 (-98%), 0.08 (-82%), and 0.02 (-96%) for the `reg`, `study-mod`, and `var-mod` hypotheses, respectively. The best Hits@10 scores were 0.01 (-98%), 0.16 (-69%) and 0.05 (-92%). Appendix A.1.2 provides additional details on the results for the subsymbolic models. We provide additional information in the `experiments` folder of our code.

6.4.4 LLM-based Method

For each *giv*, we prompted the LLM with tabular data of possible hypotheses and asked the LLM to output the top 5 hypotheses, still in tabular format. In our prompt, we ask for coherence as a ranking criterion, although we acknowledge that coherence can be a broad concept. Future work could refine this by incorporating more precise definitions. We used *gpt-3.5-turbo-0125*, and 4 times *gpt-4o* when the prompt was too long. There were 58, 58 and 18 *givs* for the `reg`, `study-mod` and `var-mod` hypothese respectively.

To check for hallucinations, we define consistency as the percentage of lines in the LLM-generated subset that exactly match a line from the input csv. For the `reg`, `study-mod`, and `var-mod` hypotheses, the outputs were consistent in 93% (54/58), 97% (56/58), and 85% (15/18) cases. Of the 9 inconsistent cases, 7 were formatting errors in which the LLM returned templated hypotheses. In only 2 cases did the output contain rows that did not directly correspond to the input, though these remained coherent with the hypotheses described in Section 6.3. Therefore, the LLM did not hallucinate new knowledge and provided reliable content.

To evaluate the hypotheses, we compared them with the findings of studies in CoDa, using a tailored evaluation approach to account for the unsupervised nature of the method. Figure 6.5 presents a histogram and scatter plot of hypothesis accuracy, defined as the ratio of `cc:Observation` supporting a hypothesis to the total of supporting and refuting observations (where refutations include negative or null findings). The LLM predominantly generates hypotheses that are misaligned with the literature, as evidenced by a high number of hypotheses with zero accuracy scores (`reg`: 112, `study-mod`: 137, `var-mod`: 22). However, some hypotheses achieve perfect alignment (scores of one: 27, 47, and 5, respectively), demonstrating that the LLM occasionally aligns well with existing evidence. In particular, extreme accuracy scores (both 0 and 1) are typically supported by fewer than 20 studies, underscoring the challenges of generalization and the presence of contradictions in the literature. This suggests that while the LLM can identify a few well-supported hypotheses, it struggles to consistently capture broader trends. Appendix A.1.3 provides more details on the prompts.

6.5 User Study

We describe our user study to answer RQ3 on comparing AI-generated hypotheses with those generated by experts.

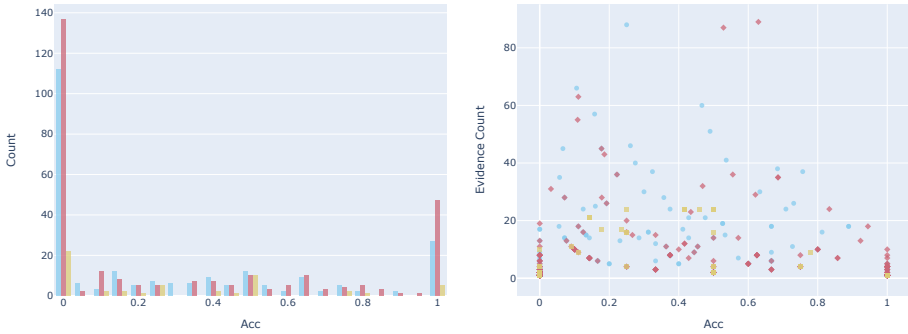

Figure 6.5: Left: distribution of accuracy scores. Right: scatter plot of accuracy scores along with the number of observations (evidence) supporting the score. Blue: `reg`, red: `study-mod`, yellow: `var-mod`.

Description. We gathered 5 domain experts for our user study, which is reasonable given the niche domain that required deep familiarity with both cooperation research and CoDa. The participants, aged 18 to 64, were based in Asia, Europe, and the United States of America. We asked participants to rate hypotheses for the 3 types of hypotheses, `reg`, `study-mod` and `var-mod`, and on 2 distinct *givs* for each type of hypothesis. For each combination of hypothesis type and *giv*, there were asked to: (1) generate the top 5 hypotheses manually; (2) score the 6 hypotheses generated by our AI methods; (3) rank the 11 hypotheses $(5+6)$ altogether. For task (2), we selected the best two hypotheses for each AI method and randomly shuffled them. Hypotheses were scored on a discrete scale from 1 to 5. In total, each participant manually generated 30 hypotheses $(3 \cdot 2 \cdot 5)$, scored 36 AI-generated hypotheses $(3 \cdot 2 \cdot 6)$ and ranked 66 hypotheses $(3 \cdot 2 \cdot 11)$. Participants were able to distinguish their own hypotheses from the AI-generated ones but not between the different AI methods. Participants scored and ranked hypotheses based on their relevance and perceived interestingness for further investigation. The survey[2] we created is openly available[3].

Results. We provide an overview of the hypotheses generated by experts and compare them to the AI-generated methods. Regarding the **human-generated hypotheses**, participants rated their hypotheses with a score of 3.9 in terms of relevance or interestingness to investigate. Of the 150 manually generated

[2]We used Google Forms: `https://workspace.google.com/intl/en/products/forms/`.

[3]`https://bit.ly/coda-user-study-pdf`

hypotheses ($30 \cdot 5$), the number of similar hypotheses for the `reg` category was 3, 1, and 2, among 5, 4, and 3 participants, respectively. For the `var-mod`, `study-mod` and `reg` categories, the number of similar hypotheses across 3 users was 5, 3, and 2, respectively. This shows the high diversity of generated hypotheses, as well as various interests from researchers, therefore highlighting the demanding task of generating a good, both relevant and interesting.

Table 6.4: Mean ranks and scores for all hypotheses. A higher score is better, a lower rank is better. Bold: best metric, underlined: best second metric.

Model	`reg`		`study-mod`		`var-mod`	
	Score	Rank	Score	Rank	Score	Rank
human	**3.8**	**5.0**	**3.6**	**4.6**	**4.0**	**5.6**
classification	<u>3.5</u>	<u>6.5</u>	1.6	7.9	<u>3.4</u>	6.7
anyburl	3.0	<u>6.5</u>	1.8	7.5	3.1	<u>5.7</u>
llm	2.0	7.6	<u>2.8</u>	<u>6.0</u>	2.9	6.7

For the **comparison between AI-generated hypotheses and those generated by experts**, the quadratic Cohen's kappa, which quantitatively measures inter-annotator agreement, for the AI hypotheses scoring and the hypotheses ranking was of 0.29 and 0.015, respectively, which shows that characteristics such as relevance and interestingness are subjective. Table 6.4 shows the mean ranks and scores for all hypotheses. While human-generated hypotheses consistently achieved the highest scores and best ranks, AI methods also demonstrated their potential. AnyBURL achieved an almost similar ranking for the `var-mod` hypotheses ($+1.8\%$), although with a lower average scoring. The classification model performs competitively for the `reg` hypotheses for scoring (-7.9%). The LLM model is the second best one for the `study-mod` hypotheses, although with a bigger difference in scoring (-22%). The supervised and unsupervised methods are, therefore, complementary. Appendix A.1.4 provides details on the distribution of users across hypotheses and of scores.

Averaging results does not account for individual variations across participants in the user study. Figure 6.6 shows the distribution of the hypotheses' ranks across the different same models, where "human" represents the manually crafted hypotheses. The distribution is spread across each model, with some human-generated hypotheses ranked first and last, and the same goes for AI-generated hypotheses. This is in line with our previous intuition that there are individual differences in perceived relevance and interestingness of the hypotheses. We also observed that one participant consistently placed human-generated hypotheses in their top 3 rankings, while another participant consistently put AI-generated hypotheses in their top 3 rankings.

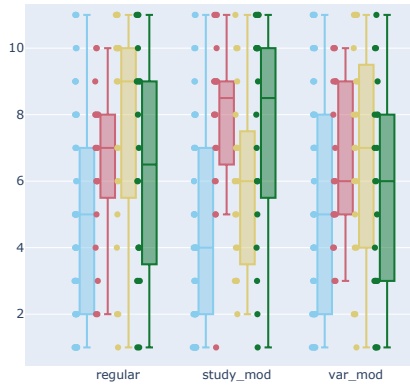

Figure 6.6: Distribution of hypotheses' mean rank across type of hypothesis and models. Blue: human, green: AnyBURL, red: classification, yellow: LLM. A lower rank is better.

The variability in human rankings highlights the subjective nature of the field, where using metrics alone to rank hypotheses can be challenging. However, it also illustrates the value of AI methods, which can provide diverse insights that traditional evaluation might miss. Evaluating participant preferences for diversity versus accuracy is complex, as judgments of relevance and interestingness are often subjective. For example, an accurate hypothesis may be less valued if it is too similar to the existing literature, while a diverse hypothesis may lack immediate appeal to researchers.

Summary. AI-generated hypotheses provide significant value by offering diverse, efficiently produced options that complement human-generated ideas. Although they may not consistently outperform human hypotheses, their efficiency and reliability make them a valuable tool for enhancing hypothesis generation and supporting social scientists in their daily work. Importantly, our methods can uncover sensible and investigable hypotheses that domain experts might overlook, broadening the scope of exploration. Referring back to Figure 6.1, which outlines our envisioned human-centric process to generate meta-reviews, AI and expert-generated hypotheses are typically integrated in Step 1. In Step 2, users select hypotheses for further exploration, as demonstrated when participants ranked hypotheses in our study. The fact that AI-generated hypotheses often rank among the top three indicates their potential to be selected for deeper investigation, reinforcing their utility within this collaborative framework.

Feedback. The participants underlined that hypothesis discovery was a challenging task. Ratings of clarity of the instructions varied between participants, but overall the task was perceived as difficult. Furthermore, one participant did not agree with the `study-mod` and `var-mod` templates, emphasising even more the challenge of finding a consensus for generating hypotheses.

6.6 Conclusion

To better assist social scientists in their daily tasks, we use knowledge-enriched AI methods for hypothesis generation as an alternative to manual standard processes. We focus on the case of human cooperation, a critical aspect of social behavior with extensive implications for economic activities and social policies [41, 154, 304]. We first collaborated with experts to design three template hypotheses. We then implemented three categories of models for automated hypothesis generation: a classification system based on decision trees, a link prediction system from graph-based methods, and an LLM system based on ontologically constrained prompting. Lastly, we conducted a user study with domain experts to compare AI-generated hypotheses with those generated by experts. We find that AI-generated hypotheses, though not always surpassing human-generated ones, provide diverse, efficiently produced options that effectively complement and enhance the hypothesis generation process. We envision integrating these AI-generated hypotheses into a digital assistant that facilitates hypothesis and meta-review generation. We have already worked with domain experts to provide an interface for the social scientists on automated meta-review generation that summarizes the findings of meta-analyses [33].

For hypothesis generation in other domains, we believe that the most affected RQ will be RQ1 on the requirements to formulate hypotheses, while RQ2 and RQ3 would remain similar. As in this work, cooperation with domain experts will be necessary to derive valuable insights from the initial data and translate them into structured knowledge. The low model complexity of our approach can be advantageous when dealing with limited data, ensuring interpretability and trainability.

In future work, our aim is to refine the design of the hypothesis by collaborating with experts to better filter meaningful moderators [177], and explore more complex hypotheses. We will also experiment with advanced methods such as hypergraph link prediction [164, 371]. Additionally, we plan to shift from a human-in-the-pipeline approach to a human-in-the-loop approach, allowing AI methods to incorporate user feedback. Future research could also explore other research fields within social science and beyond. Lastly, our user study can introduce biases, as participants ranked both their own hypotheses

and AI-generated ones, potentially favoring their own. Future work could explore independent evaluations to reduce self-assessment bias and qualitative methods, such as interviews, to provide deeper insights.

7 Conclusions

This thesis presented new algorithms, methods and results on using Knowledge Graphs (KGs) to enable narrative-like functions such as explanation, hypothesis generation, and discourse analysis. Our aim has been to explore how structured data, models and representations can support computational narratives across different domains and data types, bridging symbolic and statistical approaches. We find that KGs provide a flexible and expressive foundation for modeling narrative elements such as events, actors, and temporal relations, and that they can be adapted to support diverse tasks, from historical question answering to the analysis of social media discourse and hypotheses generation in the social sciences.

In this chapter, we first revisit the three use cases, before responding to our research questions and concluding with outlook on future directions.

7.1 Recap of the Use Cases

The three use cases (Chapters 3 to 6) illustrate the broad applicability of our approach, demonstrating how it can be adapted and instantiated to meet the specific demands of different data environments and narrative objectives.

Historical data (Generic memory narratives, Chapters 3 and 4)

Example. *"In which events were Jean-Baptiste Jourdan and Joseph Bonaparte both involved during the French Revolution?"* This question requires more than simple fact generation: it demands querying the KG to extract verified information about each individual's involvement in events. The system then assembles this data into a coherent narrative that highlights their over-

lapping roles and interactions. In this context, maintaining groundedness is essential to avoid purely generative or speculative answers.

Context and Data. This use case investigates how narratives can be generated from large-scale, generic KGs like Wikidata and DBpedia. These KGs contain millions of triples describing historical facts, events, and entities. Here, the constructed memory is generic, aiming to compress large volumes of open knowledge into coherent explanatory narratives. Although this memory is generic, it still carries the bias of the historian or the input source [310].

Narrative Function. The goal is to support explanatory narratives and question answering in a historical context, e.g. answering *"Why did the U.S. enter World War II?"* or *"What were the key events of the Cold War?"* This involves organizing a collection of facts into a causal or temporal narrative that provides clarity and interpretability.

Applying the Approach

- KG Representation: it was modeled around the 5Ws (Who, What, When, Where, Why), allowing event-centric representation.
- KG Construction
  - Content Retrieval: a semantically informed best-first search algorithm was developed to traverse the KG starting from a focal entity, prioritizing semantically meaningful paths while pruning irrelevant ones.
  - KG Population: rule-based schema conversion and information extraction techniques were applied to structure retrieved facts into a coherent, event-focused graph.
  - KG Validation: the constructed subgraphs were evaluated by comparing them to ground truth graphs.
- Narrative Retrieval
  - Graph Querying: SPARQL queries were used to extract relevant event subgraphs associated with a given topic or entity.
  - Transformation: an LLM was leveraged to transform the structured data into fluent, natural-language narratives.

Domain-Specific Research Challenges and Contributions. The primary challenge was scaling narrative generation over large, generic KGs where the number of possible connections is exponential. The solution involved designing an **efficient graph traversal method to guide content selection semantically**. Unlike more curated or domain-specific datasets, the open nature of these KGs required extensive filtering and heuristic control to maintain narrative relevance and coherence.

Social media (Personal memory narratives, Chapter 5)

Example. *"We see the end of the trauma created by our brutal system of race and gender inequality"*. This reflects a personal narrative in public discourse, where an individual frames a view on social conditions. In the context of the social media use case, such narratives are analyzed to understand how different group-level characteristics such as gender, race, or ideological affiliation influence the way users frame social issues like inequality. By extracting and structuring these narratives in a KG, it becomes possible to compare how similar events or topics are discussed across user groups.

Context and Data. Social media narratives aim to make sense of present events, often prioritizing persuasive impact over factual accuracy. Their rapid spread is driven less by objective reasoning and more by emotional resonance and social relevance [310]. This use case focused on discourse surrounding inequality on Twitter. The constructed KG reflects a personal memory of public discourse, capturing how individuals and groups frame social issues online.

Narrative Function. The goal is discourse analysis, specifically to understand how different framings and rhetorical strategies emerge over time. Narratives here help to observe and contextualize beliefs, revealing implicit ideological positions and communicative patterns.

Applying the Approach

- KG Representation: it was derived from competency questions that formalize the types of analysis the KG should support (e.g., "How is inequality framed by different user groups?").
- KG Construction
  - Content Retrieval: the Twitter API was used to collect relevant tweets and metadata using keyword-based filters.

  - KG Population: rule-based methods were employed to extract features from both the tweet text and metadata, and populate the graph according to a custom ontology.
  - KG Validation: SHACL shapes were applied to check that the populated graph conformed to the ontology.

- Narrative Retrieval
  - Graph Querying: SPARQL queries were used to extract relevant content associated with a given use case.
  - Transformation: Instead of textual storytelling, outputs were tabular data listing the instantiated patterns.

Domain-Specific Research Challenges and Contributions. The main challenge was designing an ontology expressive enough to capture rhetorical and social aspects of discourse. In contrast to historical data, the input was relatively easy to gather, but required careful modeling to ensure it could answer nuanced questions. The solution involved the **design and release of an ontology and a KG**. The result is a more personal and temporal narrative memory, reflecting a dynamic, ongoing and sometimes contradicting conversation rather than static historical facts.

Social sciences (Domain-specific narratives, Chapter 6)

Example. *"Cooperation is significantly higher when the individual difference level is low compared to when it is high."* Here, individuals participate in public goods games, and the goal is to compare how cooperation levels vary across groups with different internal compositions. The individual difference level refers to variations between individuals in traits, behaviors, or responses such as personality or beliefs that can influence how they perceive, interpret, or act in a given context [359]. In the example, groups with lower variability, i.e., more similar individuals, are hypothesized to exhibit higher cooperation, potentially due to aligned expectations or strategies. In our use case, a hypothesis is formally expressed through ontological constraints, and tested by analyzing aggregated outcomes from multiple experimental studies.

Context and Data. This use case focuses on meta-analytic data from the social sciences, particularly studies on cooperation games across different cultural contexts. The data originates from a curated, structured KG built with the help of domain experts. The data represents a domain-specific memory, emphasizing formal, explicit relationships and variables. This use case seeks

to understand how group-level features impact cooperation, through hypotheses collectively built but rarely formalized. Inspired by clinical narratives that investigate the outcome of treatments effects through clinical trials [310], it attempts to bring structure to theory-driven social data.

Narrative Function. The task is hypothesis generation, i.e. producing hypotheses that reflect underlying patterns in social behavior and are aligned with expert reasoning. Unlike historical or social media narratives, these are templated, scientific narratives that aim to advance theoretical understanding.

Applying the Approach

- KG Representation: in close collaboration with domain experts, the elements were identified and based on theoretical constructs (e.g., cultural traits, game outcomes).
- KG Construction
  - Content Retrieval: structured datasets were accessed using SPARQL queries, focusing on relevant studies and variables.
  - KG Population: a graph was constructed using rules and including blank nodes to represent hypotheses.
  - KG Validation: discussions with experts ensured that both the KG structure and its content were semantically coherent.
- Narrative Retrieval
  - Graph Querying: machine learning, link prediction and LLM techniques were combined to explore plausible links and patterns in the graph.
  - Transformation: hypotheses were generated as templated textual statements.

Domain-Specific Research Challenges and Contributions. The main challenge was ensuring that generated hypotheses were meaningful and useful to experts. The solution involved **ontological constraints and expert feedback throughout the process**. The resulting hypotheses help researchers think through possibilities, build new theories, and synthesize knowledge in a structured way.

These three use cases demonstrate that the approach can be instantiated across diverse domains, each involving different data types and narrative functions. These use cases contribute to answering the research questions in the next section.

7.2 Answering the Research Questions

We here elaborate on our results as we answer the main research questions of this thesis.

Research Question 1 (Representation): How to *represent* narrative-like structures as knowledge graphs?

Narrative-like structures can be represented as knowledge graphs by identifying key requirements such as events, agents, and their relationships, and by selecting or designing ontological models that support narrative coherence and structure (Chapter 2). While individual components are relatively straightforward to formalize, representing their interconnections is more challenging and requires more careful ontological design. To address this, we propose practical guidelines for choosing suitable structured formats (Chapter 2). We then apply them across three use cases in this thesis ("KG Representation" outlined in Section 7.1): modeling historical narratives (Chapters 3 and 4), analyzing social media discourse (Chapter 5), and supporting hypothesis generation in the social sciences (Chapter 6).

Research Question 2 (Construction): How can event-centric KGs be *constructed* from heterogeneous data sources across domains?

KGs can be constructed through a two-step process: targeted content retrieval ("Content Retrieval" from 7.1) followed by rule-based graph construction ("KG Population" and "KG Validation" from 7.1). Each use case applied domain-specific strategies to identify and select relevant data (Chapters 3, 5 and 6). Graphs were then built using rule-based methods adapted to the structure and semantics of each domain. Across all cases, this modular approach proved effective for creating event-centric KGs from diverse and heterogeneous sources (Chapters 3, 4, 5, and 6).

Research Question 3 (Evaluation): How *effective* are structured narrative representations for supporting reasoning tasks, such as question answering or hypothesis generation?

Structured narrative representations can effectively support both automated and human-centered reasoning tasks, drawing on the retrieval approaches discussed in Section 7.1 (“Narrative Retrieval”), and more specifically in the historical and social science domains.

Intrinsic (quantitative) evaluations use task-specific metrics across the use cases. ChronoGrapher, based on structured representations, outperforms baselines in retrieving sub-event subgraphs, also performing well on generating KGs (Chapter 3). Encoding narrative properties and sub-events generally enhances link prediction, though modeling roles may reduce performance; reification yields the best results, likely due to contextual modeling (Chapter 4). For hypothesis generation and despite overall modest results, rule-based methods outperform embeddings, and large language models sometimes produce hypotheses too closely aligned with domain literature (Chapter 6).

Extrinsic (qualitative) evaluations through user studies show that prompts enriched with event-centric KG triples improve factual accuracy in question answering without compromising succinctness (Chapter 3). Furthermore, AI-generated hypotheses enhance idea diversity and often rank among the top options for further exploration, supporting effective human-AI collaboration despite not consistently outperforming human-generated hypotheses (Chapter 6).

7.3 Outlook

This thesis has introduced methods for constructing and applying structured narrative representations across diverse domains. Building on this foundation, several avenues for future research emerge.

A central direction is the development of human-centered digital assistants that support narrative reasoning in everyday tasks. While KGs offer transparency and structure, they are often too complex for direct human use. Future systems should *decouple backend representation from user-facing interaction, using for instance natural language, visualizations, or embeddings to surface relevant information in accessible ways.* Methods such as retrieval-augmented generation and learned graph representations could bridge symbolic structure with flexible downstream use, as we start to do in Chapter 4.

Developing higher-level, interoperable narrative ontologies remains a key challenge. Addressing this would enable more unified and reusable representations across domains and syntactic formats, thus supporting more systematic evaluation of narrative structure and semantics. This speaks directly to the broader issue of how narratives should be represented, the focus of our first research question.

To scale and generalize narrative modeling, *improving the automation of KG construction* is essential, a core concern of our second research question. Current pipelines still rely heavily on ontology-aware design and manual tuning. Future work should explore more adaptive, ontology-informed models capable of generalizing across schemas and domains.

In domain-specific applications like the historical domain, social media discourse or the social sciences, structured representations should support collaborative hypothesis generation, explanation and refinement, integrating user goals, feedback, and contextual awareness. This also calls for *improved evaluation methods*, as a follow-up of our third research question. Future systems must go beyond generating outputs to actively supporting human reasoning through *explainability and alignment with users' expectations*.

Ultimately, structured narrative AI should embody more than technical accuracy: it must be transparent, goal-aware, and communicative. This calls for a shift toward systems that not only represent knowledge but also *interact with users in interpretable and human-centered ways.*

***

This thesis explored how structured knowledge can support narrative understanding, not only by storing information, but by enabling systems to explain, contextualize, and hypothesize. While much work remains, the methods presented here mark a step towards AI systems that engage more meaningfully with human reasoning, enabling narrative-aware tools to assist users in answering questions and analyzing complex phenomena.

A Appendix

A.1 Automated Hypothesis Generation

A.1.1 Classification Task

Table A.1: Search space for the hyperparameter optimisation for TransE [38], DistMult [370], ComplEx [330], and R-GCN [291]. The learning include the following values: $[0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1]$. The left part describes the parameters for the random search with 500 parameters. After filtering, the right part describes the parameters for the final search with 378 parameters. The model we used was DistMult for this final search. Dim.: embedding dimension; lr: learning rate; # of negs: number of negatives; # of epochs: number of epochs.

Phase	**1**			**2**		
Hyperparam.	**min**	**max**	**step**	**min**	**max**	**step**
Dim.	16	512	16	192	512	16
lr	0.001	0.1	*log*	0.001	0.001	-
# of negs	1	100	10	1	60	10
# of epochs	100	500	100	300	500	100

Embeddings. Table A.1 details the parameters explored during the phases: (1) Random search across a set of 500 parameters. (2) Full search on a subset of these parameters. Following the random search, we observed a significant negative correlation between the learning rate and the Mean Reciprocal Rank (MRR) score ($corr = -0.20$, $pval = 6.46 \times 10^{-6}$). Table A.2 displays the optimal values identified for each model during the random search, highlighting

that DistMult achieved the highest scores. Further investigation into results with a learning rate of 0.001 and DistMult as the model revealed that metrics improved slightly with larger embedding dimensions, fewer negative samples, and more epochs. These insights guided the parameters selected for the subsequent full search (2). After conducting the remaining 378 experiments, we found a strong negative correlation between the number of negative samples and the MRR score ($corr = -0.48$, $pval = 3.11 \times 10^{-23}$). Specifically, experiments with 1 negative sample showed a significant positive correlation between the number of epochs and the MRR score ($corr = -0.56$, $pval = 2.0 \times 10^{-6}$). No significant differences were found in embedding dimensions, thus we retained the dimension (336) that yielded the best score.

Table A.2: Optimal values identified for the random search.

Hyperparam.	**TransE [38]**	**DistMult [370]**	**ComplEx [330]**	**R-GCN [291]**
# configs	131	111	133	135
Embedding dim.	16	416	32	80
Learning rate	0.02	0.001	0.02	0.005
# of negatives	90	1	30	100
# of epochs	300	500	400	500
Hits@1	0.0957	**0.215**	0.0800	0.146
Hits@3	0.186	**0.391**	0.131	0.296
Hits@10	0.289	**0.501**	0.195	0.489
MRR	0.167	**0.320**	0.122	0.259

Model. Table A.3 shows the optimal parameters we searched for the model: we trained the model on the training set and evaluated it on the validation set. Table A.4 shows the best parameters for each hypothesis type.

Table A.3: Hyperparameter search for the classification.

Parameter	**Values**
criterion	[gini, entropy]
splitter	[best, random]
max_depth	[None, 5, 15, 20]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4]
max_features	[None, sqrt, log2, 0.5, 0.7]
max_leaf_nodes	[None, 10, 30]

Table A.4: Best parameters for the classification task, with a train/val/test split of 0.8/0.1/0.1.

Hypothesis	**Regular**	**Study-Mod**	**Var-Mod**
Best Parameters			
criterion	gini	entropy	gini
splitter	random	random	random
max_depth	20	15	5
min_samples_split	5	10	2
min_samples_leaf	2	2	2
max_features	sqrt	None	0.5
max_leaf_nodes	None	None	None

Complex Models. We compared the performance of the simple decision tree classifier with more complex classification models. For each model we compared different hyperparameters. Table A.5 shows the best accuracy on the validation set for each model, along with the corresponding accuracy on the train set and the number of experiments run. Given the similarity of results compared to the decision tree, we opted for simplicity and kept the decision tree classifier.

Table A.5: Mean ranks and scores for all hypotheses. A higher score is better, a lower rank is better. Bold: best metric, underlined: best second metric.

Model type	**Model**	`reg`			`study-mod`			`var-mod`		
		# exp.	Train	Val.	# exp.	Train	Val.	# exp.	Train	Val.
Tree	`DecisionTree`	-	0.79	0.78	-	0.80	0.80	-	0.74	0.70
NN	`MLP`	96	0.77	0.74	196	0.80	0.80	96	0.73	0.67
Neighbors	`KNeighbors`	72	0.79	0.72	72	0.81	0.75	72	0.71	0.77
Linear	`Ridge`	-	-	-	19	0.77	0.78	-	-	-
Linear	`SGDC`	96	0.77	0.76	96	0.77	0.78	96	0.74	0.72
Ensemble	`AdaBoost`	9	0.75	0.76	9	0.77	0.78	9	0.74	0.70
Ensemble	`Bagging`	100	0.79	0.78	100	0.81	0.78	100	0.76	0.74
Ensemble	`GradientBoosting`	100	0.79	0.78	100	0.80	0.80	100	0.61	0.70
Ensemble	`Random Forest`	100	0.77	0.78	100	0.81	0.80	100	0.73	0.70

A.1.2 Link Prediction Task

Symbolic method (AnyBURL). Figure A.1 shows the distribution of the support scores of the rules learned by AnyBURL. The figure on the left shows

the percentage distribution ; for the three types of hypothesis, the majority of the support scores are below 0.1. The figure on the right shows the count distribution having removed the scores below 0.1.

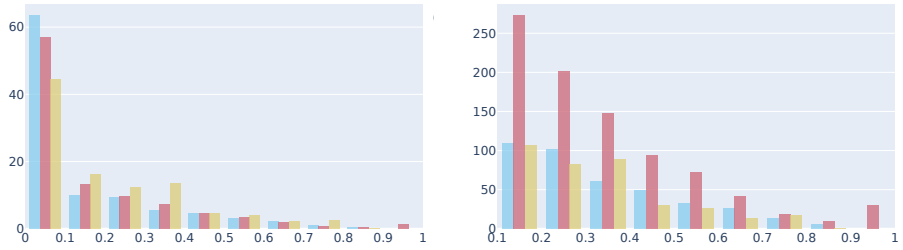

Figure A.1: Distributions of support scores of rules learned by AnyBURL. Left: percentage distribution; right: count distributes for rules whose support score is above 0.1 for more clarity. Blue: `reg`, red: `study-mod`, yellow: `var-mod`.

Figure A.2 illustrates the distribution of support scores for rules, categorized by the different antecedents on the x-axis. The scores are mostly between 0.2 and 0.4, and the rules with `cp:mod` as antecedents are the ones with the lowest support scores. Figure A.3 shows the top 3 rules learned for each type of hypothesis. All these rules are of length 1. For instance, the top rule with a score of 0.85 for the `var-mod` rule states that studies with `id:uncertaintylevel/high` as the second *sivv* has a `positiveEffectOn contributions`.

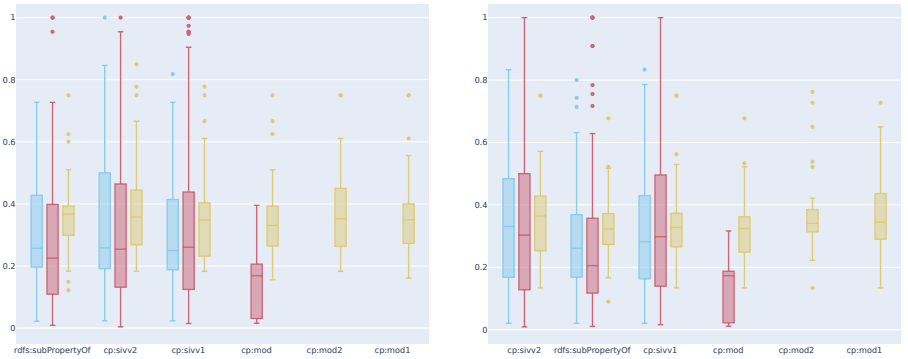

Figure A.2: Boxplot of scores for antecedents in rules. Left: positive effect, right: negative effect. Blue: `reg`, red: `study-mod`, yellow: `var-mod`.

Top 3 Rules Learned for the reg hypothesis

```
cp:hasPositiveEffectOn(X,cooperation)
<= cp:sivv2(
X,id:intergroupcomp/intergroup_prisoner's_dilemma)(1.0)
cp:hasPositiveEffectOn(X,cooperation)
<= cp:sivv2(
X,id:motivationalorientation/competitive)(0.85)
cp:hasNegativeEffectOn(X,contributions)
<= cp:sivv1(X,id:uncertaintytarget/threshold)(0.83)
```

Top 3 Rules Learned for the study-mod hypothesis

```
cp:hasNegativeEffectOn(X,cooperation)
<= rdfs:subPropertyOf(
X,cp:continuationProbabilityLevel)(1.0)
cp:hasPositiveEffectOn(X,contributions)
<= cp:sivv1(X,id:religiouslevel/high)(1.0)
cp:hasNegativeEffectOn(X,cooperation)
<= cp:sivv2(X,id:emotion/happiness)(1.0)
```

Top 3 Rules Learned for the var-mod hypothesis

```
cp:hasPositiveEffectOn(X,contributions)
<= cp:sivv2(X,id:uncertaintylevel/high)(0.85)
cp:hasPositiveEffectOn(X,cooperation)
<= cp:sivv2(X,id:decisionmaker/group)(0.78)
cp:hasPositiveEffectOn(X,cooperation)
<= cp:sivv1(X,id:decisionmaker/individual)(0.78)
```

Figure A.3: Top 3 Rules Learned for the Three Types of Hypotheses.

Table A.6: Optimal values identified for the random search for the link prediction method. T: TransE [38], D: DistMult [370], C: ComplEx [330], R: R-GCN [291].

	reg				study-mod				var-mod			
Hyperparam.	T	D	C	R	T	D	C	R	T	D	C	R
# configs	14	9	9	8	14	9	9	8	14	9	9	8
Embedding Dim.	272	192	432	512	432	496	32	160	432	496	32	304
Learning rate	0.02	0.1	0.1	0.01	0.0	0.01	0.01	0.0	0.0	0.01	0.01	0.02
# of negatives	10	1	80	20	80	10	40	100	80	10	40	70
# of epochs	100	200	200	200	300	500	300	300	300	500	300	500
Hits@1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.0	0.0
Hits@3	0.01	0.0	0.0	0.0	0.01	0.0	0.0	0.11	0.02	0.0	0.01	0.02
Hits@10	0.01	0.0	0.01	0.0	0.03	0.0	0.01	0.16	0.05	0.02	0.03	0.02
MRR	0.01	0.0	0.01	0.0	0.01	0.0	0.0	0.08	0.02	0.0	0.01	0.01

Sub-symbolic methods. Table A.6 shows the results of the best parameters for each model and hypothesis type on the validation set. Given the very low results on these models and despite the parameter optimisation, we decided to discard the models, especially since their symbolic counterpart performed much better.

A.1.3 LLM Task

Figure A.4, A.5 and A.6 shows examples of prompts we used for the `reg`, `study-mod` and `var-mod` hypotheses respectively. The only difference is in the data format and the template hypothesis.

A.1.4 User Studies

Figure A.7 shows the distribution of users across *sivs* and *moderators*, distinguished by the type of hypothesis. There is more overlap for the common *sivs* and *moderators* used across users for the generated hypotheses, which shows that there would be a higher consensus for the *interesting siv* to explore for a hypothesis. Figure A.8 shows the distribution of the hypotheses' score across the type of hypothesis and models.

Prompt:
You will be given data in .csv format, where each line can be read as a hypothesis on how humans cooperate. The csv has the following columns: ‘dependent‘, ‘giv_prop‘, ‘iv‘, ‘iv_label‘, ‘cat_t1‘, ‘cat_t1_label‘, ‘cat_t2‘ and ‘cat_t2_label‘.
Each line in the data can be templated as a hypothesis as follows: "Cooperation is significantly ‘comparative‘ when ‘iv_label‘ is ‘cat_t1_label‘ compared to when ‘iv_label‘ is ‘cat_t2_label‘.". ‘iv_label‘, ‘cat_t1_label‘ and ‘cat_t2_label‘ comes from the data ; ‘comparative‘ must be added and can only take the values ‘higher‘ or ‘lower‘.
Given the data, you must extract the subset of 5 rows that represent the most coherent hypotheses, ranked from the most coherent one. If there are less than 5 rows, you can re-order them by order of coherence. On top of the existing columns, you must add another one, ‘comparative‘ that can take the values ‘higher‘ or ‘lower‘.
The output must be in csv format.

Figure A.4: Prompt used for the `reg` hypothesis.

Prompt:
You will be given data in .csv format, where each line can be read as a hypothesis on how humans cooperate. The csv has the following columns: ‘dependent‘, ‘giv_prop‘, ‘iv‘, ‘iv_label‘, ‘cat_t1‘, ‘cat_t1_label‘, ‘cat_t2‘, ‘cat_t2_label‘, ‘mod‘, ‘mod_label‘, ‘mod_val‘ and ‘mod_val_label‘.
Each line in the data can be templated as a hypothesis as follows: "When comparing studies where ‘iv_label‘ is ‘cat_t1_label‘ and studies where ‘iv_label‘ is ‘cat_t2_label‘, cooperation is significantly ‘comparative‘ when ‘mod_label‘ is ‘mod_val_label‘ compared to when ‘mod_label‘ has another value.". ‘iv_label‘, ‘cat_t1_label‘, ‘cat_t2_label‘, ‘mod_label‘, and ‘mod_val_label‘ comes from the data ; ‘comparative‘ must be added and can only take the values ‘higher‘ or ‘lower‘.
Given the data, you must extract the subset of 5 rows that represent the most coherent hypotheses, ranked from the most coherent one. If there are less than 5 rows, you can re-order them by order of coherence. On top of the existing columns, you must add another one, ‘comparative‘ that can take the values ‘higher‘ or ‘lower‘.
The output must be in csv format.

Figure A.5: Prompt used for the `study-mod` hypothesis.

Prompt:
You will be given data in .csv format, where each line can be read as a hypothesis on how humans cooperate. The csv has the following columns: ‘dependent‘, ‘giv_prop‘, ‘iv‘, ‘iv_label‘, ‘cat_t1‘, ‘cat_t1_label‘, ‘cat_t2‘, ‘cat_t2_label‘, ‘mod‘, ‘mod_label‘, ‘mod_t1‘, ‘mod_t1_label‘, ‘mod_t2‘ and ‘mod_t2_label‘.
Each line in the data can be templated as a hypothesis as follows: "When comparing studies where ‘iv_label‘ is ‘cat_t1_label‘ and studies where ‘iv_label‘ is ‘cat_t2_label‘, cooperation from studies involving ‘mod_t1_label‘ as ‘mod_label‘ is significantly ‘comparative‘ than cooperation from studies involving based on ‘mod_t2_label‘ as ‘mod_label‘.". ‘iv_label‘, ‘cat_t1_label‘, ‘cat_t2_label‘, ‘mod_label‘, ‘mod_t1_label‘, and ‘mod_t2_label‘ comes from the data ; ‘comparative‘ must be added and can only take the values ‘higher‘ or ‘lower‘.
Given the data, you must extract the subset of 5 rows that represent the most coherent hypotheses, ranked from the most coherent one. If there are less than 5 rows, you can re-order them by order of coherence. On top of the existing columns, you must add another one, ‘comparative‘ that can take the values ‘higher‘ or ‘lower‘.
The output must be in csv format.

Figure A.6: Prompt used for the `var-mod` hypothesis.

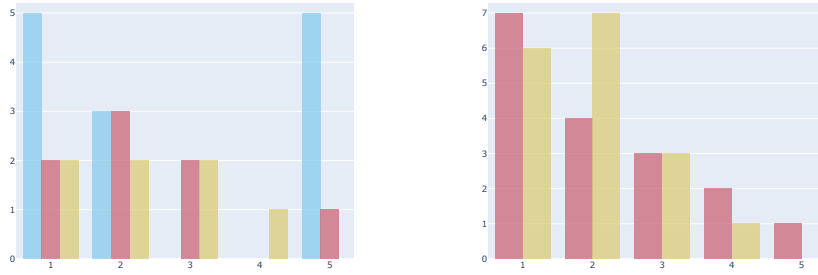

Figure A.7: Distribution of users across type of hypothesis and *sivs* (left), and across type of hypothesis and *mods* (right). Blue: `reg`, red: `study-mod`, yellow: `var-mod`.

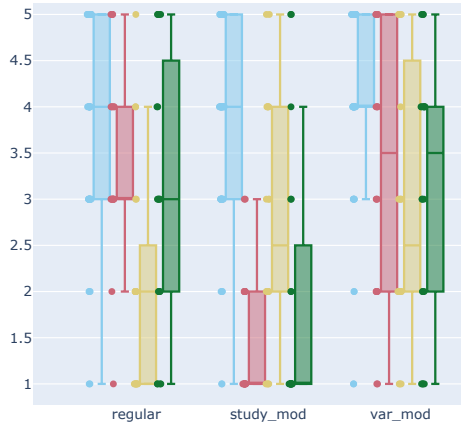

Figure A.8: Distribution of hypotheses' score across type of hypothesis and models. Blue: human, green: AnyBURL, red: classification, yellow: LLM.

References

[1] M. Addis, M. Boniface, S. Goodall, P. Grimwood, S. Kim, P. Lewis, K. Martinez, and A. Stevenson. “SCULPTEUR: Towards a New Paradigm for Multimedia Museum Information Handling”. In: *ISWC*. 2003.

[2] K. S. Aggour, A. Detor, A. Gabaldon, V. Mulwad, A. Moitra, P. Cuddihy, and V. S. Kumar. “Compound Knowledge Graph-Enabled AI Assistant for Accelerated Materials Discovery”. In: *Integrating Materials and Manufacturing Innovation* (2022).

[3] M. Alam, A. Gangemi, V. Presutti, and D. Reforgiato Recupero. “Semantic Role Labeling for Knowledge Graph Extraction from Text”. In: *Progress in AI* (2021).

[4] M. Alam, M. Kaschura, and H. Sack. “Apollo: Twitter Stream Analyzer of Trending Hashtags: A Case-study of # COVID-19.” In: *ISWC (Demos/Industry)*. 2020.

[5] A. I. Alhussain and A. M. Azmi. “Automatic Story Generation: A Survey of Approaches”. In: *ACM Computing Surveys* (2021).

[6] M. Ali, M. Berrendorf, M. Galkin, V. Thost, T. Ma, V. Tresp, and J. Lehmann. “Improving Inductive Link Prediction Using Hyper-relational Facts”. In: *ISWC*. 2021.

[7] D. Alivanistos, S. van der Bijl, M. Cochez, and F. van Harmelen. “The Effect of Knowledge Graph Schema on Classifying Future Research Suggestions”. In: *International Workshop on Natural Scientific Language Processing and Research Knowledge Graphs*. 2024.

[8] F. Alkhateeb, J.-F. Baget, and J. Euzenat. “Extending SPARQL with regular expression patterns (for querying RDF)”. In: *JWS* (2009).

[9] J. F. Allen. “Maintaining Knowledge about Temporal Intervals”. In: *Communications of the ACM* (1983).

[10] D. Alocci, J. Mariethoz, O. Horlacher, J. T. Bolleman, M. P. Campbell, and F. Lisacek. “Property Graph vs RDF Triple Store: A Comparison on Glycan Substructure Search”. In: *PLoS One* (2015).

[11] R. Angles, H. Thakkar, and D. Tomaszuk. “Mapping RDF Databases to Property Graph Databases”. In: *IEEE Access* (2020).

[12] R. Angles, H. Thakkar, and D. Tomaszuk. “RDF and Property Graphs Interoperability: Status and Issues”. In: *Alberto Mendelzon Workshop on Foundations of Data Management*. 2019.

[13] A. Antelmi, G. Cordasco, M. Polato, V. Scarano, C. Spagnuolo, and D. Yang. “A Survey on Hypergraph Representation Learning”. In: *ACM Computing Surveys* (2023).

[14] R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura. “COMM: Designing a Well-Founded Multimedia Ontology for the Web”. In: *ISWC*. 2007.

[15] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. “DBpedia: A Nucleus for a Web of Open Data”. In: *ISWC*. 2007.

[16] S. Auer, A. Oelen, M. Haris, M. Stocker, J. D’Souza, K. E. Farfar, L. Vogt, M. Prinz, V. Wiens, and M. Y. Jaradeh. “Improving Access to Scientific Literature with Knowledge Graphs”. In: *Bibliothek Forschung und Praxis* (2020).

[17] J. Baek, S. K. Jauhar, S. Cucerzan, and S. J. Hwang. “ResearchAgent: Iterative Research Idea Generation over Scientific Literature with Large Language Models”. In: *NAACL: Human Language Technologies (Long Papers)*. 2025.

[18] D. Bahler, B. Stone, C. Wellington, and D. W. Bristol. “Symbolic, Neural, and Bayesian Machine Learning Models for Predicting Carcinogenicity of Chemical Compounds”. In: *Journal of Chemical Information & Computer Sciences* (2000).

[19] J. Bakalara, T. Guyet, O. Dameron, A. Happe, and E. Oger. “An Extension of Chronicles Temporal Model with Taxonomies: Application to Epidemiological Studies”. In: *International Conference on Health Informatics*. 2021.

[20] M. Bal and C. Van Boheemen. *Narratology: Introduction to the Theory of Narrative*. 2009.

[21] C. Barrie and J. C.-t. Ho. "academictwitteR: an R package to access the Twitter Academic Research Product Track v2 API endpoint". In: *Journal of Open Source Software* (2021).

[22] V. Bartalesi, C. Meghini, and D. Metilli. "Steps Towards a Formal Ontology of Narratives Based on Narratology". In: *CMN*. 2016.

[23] T. Berners-Lee. *Linked Data - Design Issues*. URL: `https://www.w3.org/DesignIssues/LinkedData.html`.

[24] M. Berrendorf, E. Faerman, L. Vermue, and V. Tresp. "Interpretable and Fair Comparison of Link Prediction or Entity Alignment Methods". In: *International Joint Conference on Web Intelligence and Intelligent Agent Technology*. 2020.

[25] D. Beßler, R. Porzel, M. Pomarlan, M. Beetz, R. Malaka, and J. Bateman. "A Formal Model of Affordances for Flexible Robotic Task Execution". In: *ECAI*. 2020.

[26] D. Beßler, R. Porzel, M. Pomarlan, A. Vyas, S. Höffner, M. Beetz, R. Malaka, and J. Bateman. "Foundations of the Socio-Physical Model of Activities (SOMA) for Autonomous Robotic Agents 1". In: *Formal Ontology in Information Systems*. 2021.

[27] K. Beuls, P. Van Eecke, and V. S. Cangalovic. "A computational construction grammar approach to semantic frame extraction". In: *Linguistics Vanguard* (2021).

[28] I. Blin. "Building a French Revolution Narrative from Wikidata." In: *Semantic Techniques for Narrative-Based Understanding @IJCAI-ECAI*. 2022.

[29] I. Blin. "Building Narrative Structures from Knowledge Graphs". In: *ESWC (Satellite Events)*. 2022.

[30] I. Blin, L. Stork, L. Spillner, and C. Santagiustina. "OKG: A Knowledge Graph for Fine-grained Understanding of Social Media Discourse on Inequality". In: *K-CAP*. 2023.

[31] I. Blin, A. ten Teije, F. van Harmelen, and I. Tiddi. "Structured Representations for Narratives". In: *EKAW*. 2024.

[32] I. Blin, I. Tiddi, G. Spadaro, and A. ten Teije. "Automated Hypothesis Generation for Human Cooperation Studies". In: *HHAI*. 2025.

[33] I. Blin, I. Tiddi, G. Spadaro, and A. ten Teije. “Living Meta-Review Generation for Social Scientists: An Interface and A Case Study on Human Cooperation”. In: *EKAW (Posters, Demos and Workshops)*. 2024.

[34] I. Blin, I. Tiddi, and A. ten Teije. “Measuring the Impact of Narrative Complexity on Knowledge Graph Embeddings”. In: *ISWC*. 2025.

[35] I. Blin, I. Tiddi, R. van Trijp, and A. ten Teije. “ChronoGrapher: Event-centric Knowledge Graph Construction via Informed Graph Traversal”. In: *Semantic Web Journal* (2025).

[36] E. Blomqvist, V. Presutti, E. Daga, and A. Gangemi. “Experimenting with eXtreme Design”. In: *EKAW*. 2010.

[37] F. Bolanos, A. Salatino, F. Osborne, and E. Motta. “Artificial Intelligence for Literature Reviews: Opportunities and Challenges”. In: *AI Review* (2024).

[38] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. “Translating Embeddings for Modeling Multi-relational Data”. In: *NeurIPS* (2013).

[39] A. Borrego, D. Dessi, I. Hernández, F. Osborne, D. R. Recupero, D. Ruiz, D. Buscaldi, and E. Motta. “Completing Scientific Facts in Knowledge Graphs of Research Concepts”. In: *IEEE Access* (2022).

[40] E. Boschee, J. Lautenschlager, S. O’Brien, S. Shellman, J. Starz, and M. Ward. *ICEWS Coded Event Data*. Version V35. 2015. URL: `https://doi.org/10.7910/DVN/28075`.

[41] S. Bowles and H. Gintis. “A Cooperative Species: Human Reciprocity and Its Evolution”. In: *A cooperative species*. 2011.

[42] B. Boyd. *On the Origin of Stories: Evolution, Cognition, and Fiction*. 2009.

[43] F. Branch, T. Arias, J. Kennah, R. Phillips, T. Windleharth, and J. H. Lee. “Representing Transmedia Fictional Worlds through Ontology”. In: *Journal of the Association for Information Science and Technology* (2017).

[44] J. Brank, M. Grobelnik, and D. Mladenic. “A Survey of Ontology Evaluation Techniques”. In: *Conference on Data Mining and Data Warehouses*. 2005.

[45] J. G. Breslin, S. Decker, A. Harth, and U. Bojars. “SIOC: An Approach to Connect Web-Based Communities”. In: *International Journal of Web Based Communities* (2006).

[46] J. Bruner. “The Narrative Construction of Reality”. In: *Critical inquiry* (1991).

[47] L. Bucher, T. Burgard, U. S. Tran, G. M. Prinz, M. Bosnjak, and M. Voracek. “Keeping Meta-Analyses Alive and Well: A Tutorial on Implementing and Using Community-Augmented Meta-Analyses in PsychOpen CAMA”. In: *Advances in Methods and Practices in Psychological Science* (2023).

[48] C.-I. Bucur, T. Kuhn, D. Ceolin, and J. Van Ossenbruggen. “Expressing High-Level Scientific Claims with Formal Semantics”. In: *K-CAP*. 2021.

[49] A. Burton-Jones, V. C. Storey, V. Sugumaran, and P. Ahluwalia. “A Semiotic Metrics Suite for Assessing the Quality of Ontologies”. In: *Data & Knowledge Engineering* (2005).

[50] R. Butkienė, A. Šukys, L. Ablonskis, and R. Butleris. “Influence of Event Specialization Strategy on Some Aspects of Natural Language Querying Interfaces to Ontologies”. In: *IEEE Access* (2024).

[51] V. Campos, R. Campos, P. Mota, and A. Jorge. “Tweet2Story: A Web App to Extract Narratives from Twitter”. In: *ECIR*. 2022.

[52] G. Candela, P. Escobar, R. C. Carrasco, and M. Marco-Such. “Evaluating the quality of linked open data in digital libraries”. In: *Journal of Information Science* (2022).

[53] J. Cao, J. Fang, Z. Meng, and S. Liang. “Knowledge Graph Embedding: A Survey from the Perspective of Representation Spaces”. In: *ACM Computing Surveys* (2024).

[54] V. A. Carriero, M. Daquino, A. Gangemi, A. G. Nuzzolese, S. Peroni, V. Presutti, and F. Tomasi. “The Landscape of Ontology Reuse Approaches”. In: *Applications and Practices in Ontology Design, Extraction, and Reasoning*. 2020.

[55] V. A. Carriero, A. Gangemi, M. L. Mancinelli, L. Marinucci, A. G. Nuzzolese, V. Presutti, and C. Veninata. “ArCo: The Italian Cultural Heritage Knowledge Graph”. In: *ISWC*. 2019.

[56] T. Caselli, E. Hovy, M. Palmer, and P. Vossen. *Computational Analysis of Storylines: Making Sense of Events*. 2021.

[57] G. Castañé, A. Dolgui, N. Kousi, B. Meyers, S. Thevenin, E. Vyhmeister, and P.-O. Östberg. “The ASSISTANT project: AI for high level decisions in manufacturing”. In: *International Journal of Production Research* (2023).

[58] Š. Čebirić, F. Goasdoué, H. Kondylakis, D. Kotzinos, I. Manolescu, G. Troullinou, and M. Zneika. "Summarizing Semantic Graphs: A Survey". In: *The Very Large Data Bases Journal* (2019).

[59] I. Cervesato, M. Franceschet, and A. Montanari. "A Guided Tour through Some Extensions of the Event Calculus". In: *Computational Intelligence* (2000).

[60] L. Chai, L. Tu, X. Wang, and Q. Su. "Hypergraph modeling and hypergraph multi-view attention neural network for link prediction". In: *Pattern Recognition* (2024).

[61] D. Chanin. "Open-source Frame Semantic Parsing". In: *arXiv preprint arXiv:2303.12788* (2023).

[62] V. Chaudhri, C. Baru, N. Chittar, X. Dong, M. Genesereth, J. Hendler, A. Kalyanpur, D. Lenat, J. Sequeda, D. Vrandečić, et al. "Knowledge graphs: introduction, history and, perspectives". In: *AI Magazine* (2022).

[63] D. Chen, A. Fisch, J. Weston, and A. Bordes. "Reading Wikipedia to Answer Open-Domain Questions". In: *ACL (Long Papers)*. 2017.

[64] E. Chen, K. Lerman, E. Ferrara, et al. "Tracking Social Media Discourse About the COVID-19 Pandemic: Development of a Public Coronavirus Twitter Data Set". In: *Journal of Medical Internet Research Public Health and Surveillance* (2020).

[65] J. Chen, H. Lin, X. Han, and L. Sun. "Benchmarking Large Language Models in Retrieval-Augmented Generation". In: *AAAI*. 2024.

[66] Q. Chen and A. Crooks. "Analyzing the Vaccination Debate in Social Media Data Pre- and Post-COVID-19 Pandemic". In: *International Journal of Applied Earth Observation and Geoinformation* (2022).

[67] Y. Chen, H. Sack, and M. Alam. "Analyzing social media for measuring public attitudes toward controversies and their driving factors: a case study of migration". In: *Social Network Analysis and Mining* (2022).

[68] Z. Chen, X. Wang, C. Wang, and J. Li. "Explainable Link Prediction in Knowledge Hypergraphs". In: *CIKM*. 2022.

[69] Z. Chen, X. Wang, C. Wang, and Z. Li. "PosKHG: A Position-Aware Knowledge Hypergraph Model for Link Prediction". In: *Data Science and Engineering* (2023).

[70] G. Cheng, Y. Zhang, and Y. Qu. “Explass: Exploring Associations between Entities via Top-K Ontological Patterns and Facets”. In: *ISWC*. 2014.

[71] K. Cheng, Z. Li, C. Li, R. Xie, Q. Guo, Y. He, and H. Wu. “The Potential of GPT-4 as an AI-Powered Virtual Assistant for Surgeons Specialized in Joint Arthroplasty”. In: *Annals of Biomedical Engineering* (2023).

[72] C. Chung, J. Lee, and J. J. Whang. “Representation Learning on Hyper-Relational and Numeric Knowledge Graphs with Transformers”. In: *KDD*. 2023.

[73] T. Collins, P. Mulholland, and Z. Zdrahal. “Semantic Browsing of Digital Collections”. In: *ISWC*. 2005.

[74] L. A. Cox Jr. “An AI assistant to help review and improve causal reasoning in epidemiological documents”. In: *Global Epidemiology* (2024).

[75] Y. Cui, Z. Sun, and W. Hu. “A Prompt-Based Knowledge Graph Foundation Model for Universal In-Context Reasoning”. In: *NeurIPS* (2024).

[76] R. Cyganiak, D. Wood, M. Lanthaler, G. Klyne, J. J. Carroll, and B. McBride. *RDF 1.1 Concepts and Abstract Syntax*. W3C Recommendation. Feb. 2014.

[77] M. D’Arcy, T. Hope, L. Birnbaum, and D. Downey. “MARG: Multi-Agent Review Generation for Scientific Papers”. In: *arXiv preprint arXiv:2401.04259* (2024).

[78] R. Damiano and A. Lieto. “Ontological Representations of Narratives: a Case Study on Stories and Actions”. In: *CMN*. 2013.

[79] R. Damiano, V. Lombardo, and A. Pizzo. “The Ontology of Drama”. In: *Applied Ontology* (2019).

[80] S. Das, J. Srinivasan, M. Perry, E. I. Chong, and J. Banerjee. “A Tale of Two Graphs: Property Graphs as RDF in Oracle”. In: *Extending Database Technology*. 2014.

[81] D. Daza, M. Cochez, and P. Groth. “Inductive Entity Representations from Text via Link Prediction”. In: *WWW*. 2021.

[82] V. de Boer, J. van Doornik, L. Buitinck, M. Marx, T. Veken, and K. Ribbens. “Linking the Kingdom: Enriched access to a historiographical text”. In: *K-CAP*. 2013.

[83] R. de Haan, I. Tiddi, and W. Beek. “Discovering Research Hypotheses in Social Science Using Knowledge Graph Embeddings”. In: *ESWC*. 2021.

[84] V. De Boer, J. Oomen, O. Inel, L. Aroyo, E. Van Staveren, W. Helmich, and D. De Beurs. “DIVE into the Event-Based Browsing of Linked Historical Media”. In: *JWS* (2015).

[85] F. De Paoli, S. Berardelli, I. Limongelli, E. Rizzo, and S. Zucca. “VarChat: The Generative AI Assistant for the Interpretation of Human Genomic Variations”. In: *Bioinformatics* (2024).

[86] L. De Vocht, S. Coppens, R. Verborgh, M. Vander Sande, E. Mannens, and R. Van de Walle. “Discovering Meaningful Connections between Resources in the Web of Data.” In: *Workshop on Linked Data on the Web @WWW*. 2013.

[87] G. Del Mondo, P. Peng, J. Gensel, C. Claramunt, and F. Lu. “Leveraging Spatio-Temporal Graphs and Knowledge Graphs: Perspectives in the Field of Maritime Transportation”. In: *International Society for Photogrammetry and Remote Sensing Journal of Geo-Information* (2021).

[88] L. Derczynski, D. Maynard, G. Rizzo, M. Van Erp, G. Gorrell, R. Troncy, J. Petrak, and K. Bontcheva. “Analysis of Named Entity Recognition and Linking for Tweets”. In: *Information Processing & Management* (2015).

[89] D. Dessí, F. Osborne, D. R. Recupero, D. Buscaldi, and E. Motta. “SCICERO: A Deep Learning and NLP Approach for Generating Scientific Knowledge Graphs in the Computer Science Domain”. In: *Knowledge-Based Systems* (2022).

[90] D. Dessí, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, and E. Motta. “CS-KG: A Large-Scale Knowledge Graph of Research Entities and Claims in Computer Science”. In: *ISWC*. 2022.

[91] D. Dessı, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, E. Motta, and H. Sack. “AI-KG: an Automatically Generated Knowledge Graph of Artificial Intelligence”. In: *ISWC*. 2020.

[92] S. Di, Q. Yao, and L. Chen. “Searching to Sparsify Tensor Decomposition for N-ary Relational Data”. In: *WWW*. 2021.

[93] D. Dimitrov, E. Baran, P. Fafalios, R. Yu, X. Zhu, M. Zloch, and S. Dietze. “TweetsCOV19 - A Knowledge Base of Semantically Annotated Tweets about the COVID-19 Pandemic”. In: *CIKM*. 2020.

[94] M. Doerr. “The CIDOC Conceptual Reference Module: An Ontological Approach to Semantic Interoperability of Metadata”. In: *AI magazine* (2003).

[95] M. Doerr, C. Bekiari, P. LeBoeuf, and B. nationale de France. “FRBRoo, a Conceptual Model for Performing Arts”. In: *Annual Conference of CIDOC*. 2008.

[96] M. Doerr, S. Gradmann, S. Hennicke, A. Isaac, C. Meghini, and H. Van de Sompel. “The Europeana Data Model (EDM)”. In: *International Federation of Library Associations and Institutions World Library and Information Congress*. 2010.

[97] A. Duque-Ramos, J. T. Fernández-Breis, R. Stevens, and N. Aussenac-Gilles. “OQuaRE: A SQuaRE-based Approach for Evaluating the Quality of Ontologies”. In: *Journal of Research and Practice in Information Technology* (2011).

[98] S. Egami, K. Matsushita, T. Ugai, and K. Fukuda. “Comparison of Metadata Representation Models for Knowledge Graph Embeddings”. In: *arXiv preprint arXiv:2503.21804* (2025).

[99] S. Egami, T. Ugai, M. Oota, K. Matsushita, T. Kawamura, K. Kozaki, and K. Fukuda. “RDF-star2Vec: RDF-star Graph Embeddings for Data Mining”. In: *IEEE Access* (2023).

[100] A. Ekin, A. M. Tekalp, and R. Mehrotra. “Integrated Semantic-syntactic Video Modeling for Search and Browsing”. In: *IEEE Transactions on Multimedia* (2004).

[101] R. Eschauzier, R. Taelman, and R. Verborgh. “How Does the Link Queue Evolve during Traversal-Based Query Processing?” In: *Storing, Querying and Benchmarking Knowledge Graphs and Managing the Evolution and Preservation of the Data Web @ISWC*. 2023.

[102] P. Fafalios, V. Iosifidis, E. Ntoutsi, and S. Dietze. “TweetsKB: A Public and Large-Scale RDF Corpus of Annotated Tweets”. In: *ESWC*. 2018.

[103] G. Fakih and P. Serrano-Alvarado. “A Survey on SPARQL Query Relaxation under the Lens of RDF Reification”. In: *SWJ* (2024).

[104] H. Fan, F. Zhang, Y. Wei, Z. Li, C. Zou, Y. Gao, and Q. Dai. “Heterogeneous Hypergraph Variational Autoencoder for Link Prediction”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

[105] L. Fang, A. D. Sarma, C. Yu, and P. Bohannon. “REX: Explaining Relationships between Entity Pairs”. In: *Very Large Data Bases Endowment* (2011).

[106] B. Fatemi, P. Taslakian, D. Vazquez, and D. Poole. “Knowledge Hypergraphs: Prediction Beyond Binary Relations”. In: *IJCAI*. 2021.

[107] S. Fathalla, S. Vahdati, C. Lange, and S. Auer. “SEO: A Scientific Events Data Model”. In: *ISWC*. 2019.

[108] D. Fernàndez-Cañellas, J. Espadaler, D. Rodriguez, B. Garolera, G. Canet, A. Colom, J. M. Rimmek, X. Giro-i-Nieto, E. Bou, and J. C. Riveiro. “VLX-stories: Building an Online Event Knowledge Base with Emerging Entity Detection”. In: *ISWC*. 2019.

[109] C. J. Fillmore and C. F. Baker. “Frame Semantics for Text Understanding”. In: *WordNet and Other Lexical Resources Workshop @NAACL*. 2001.

[110] E. Filtz, M. Navas-Loro, C. Santos, A. Polleres, and S. Kirrane. “Events Matter: Extraction of Events from Court Decisions”. In: *Legal Knowledge and Information Systems*. 2020.

[111] M. A. Finlayson. *CMN*. 2012. URL: `https://narrative.csail.mit.edu/cmn12/abstracts.pdf`.

[112] M. A. Finlayson, B. Fisseni, B. Löwe, C. Meister, R. Gerrig, I. Mani, J. C. Bahamón, R. M. Young, M. Bhatt, J. Suchan, et al. *CMN*. 2013.

[114] M. A. Finlayson, W. Richards, and P. H. Winston. “Computational Models of Narrative: Review of a Workshop”. In: *AI Magazine* (2010).

[115] V. Fionda and G. Pirrò. “Explaining Graph Navigational Queries”. In: *ESWC*. 2017.

[116] V. Fionda, G. Pirrò, and M. P. Consens. “Querying Knowledge Graphs with Extended Property Paths”. In: *SWJ* (2019).

[117] V. Fionda, G. Pirrò, and C. Gutierrez. “NautiLOD: A Formal Language for the Web of Data Graph”. In: *ACM Transactions on the Web* (2015).

[118] A. Fokkens, A. Soroa, Z. Beloki, N. Ockeloen, G. Rigau, W. R. Van Hage, and P. Vossen. “NAF and GAF: Linking Linguistic Annotations”. In: *Workshop on Interoperable Semantic Annotation*. 2014.

[119] A. R. Francois, R. Nevatia, J. Hobbs, R. C. Bolles, and J. R. Smith. “VERL: An Ontology Framework for Representing and Annotating Video Events”. In: *IEEE Multimedia* (2005).

[120] M. Galkin, P. Trivedi, G. Maheshwari, R. Usbeck, and J. Lehmann. "Message Passing for Hyper-Relational Knowledge Graphs". In: *EMNLP*. 2020.

[121] M. Galkin, X. Yuan, H. Mostafa, J. Tang, and Z. Zhu. "Towards Foundation Models for Knowledge Graph Reasoning". In: *New Frontiers in Graph Learning @NeurIPS*. 2023.

[122] A. Gangemi, M. Alam, L. Asprino, V. Presutti, and D. R. Recupero. "Framester: A Wide Coverage Linguistic Linked Data Hub". In: *European Knowledge Acquisition Workshop*. 2016.

[123] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. "Sweetening Ontologies with DOLCE". In: *EKAW*. 2002.

[124] A. Gangemi and P. Mika. "Understanding the Semantic Web through Descriptions and Situations". In: *On the Move to Meaningful Internet Systems*. 2003.

[125] A. Gangemi, V. Presutti, D. Reforgiato Recupero, A. G. Nuzzolese, F. Draicchio, and M. Mongiovı. "Semantic Web Machine Reading with FRED". In: *SWJ* (2017).

[126] D. Garijo, S. Fakhraei, V. Ratnakar, Q. Yang, H. Endrias, Y. Ma, R. Wang, M. Bornstein, J. Bright, Y. Gil, et al. "Towards Automated Hypothesis Testing in Neuroscience". In: *Very Large Data Bases Workshop on Data Management and Analytics for Medicine and Healthcare*. 2019.

[127] M. Garnelo and M. Shanahan. "Reconciling Deep Learning with Symbolic Artificial Intelligence: Representing Objects and Relations". In: *Current Opinion in Behavioral Sciences* (2019).

[128] Y. Geng, J. Chen, J. Z. Pan, M. Chen, S. Jiang, W. Zhang, and H. Chen. "Relational Message Passing for Fully Inductive Knowledge Graph Completion". In: *International Conference on Data Engineering*. 2023.

[129] J. Geurts, S. Bocconi, J. Van Ossenbruggen, and L. Hardman. "Towards Ontology-driven Discourse: From Semantic Graphs to Multimedia Presentations". In: *ISWC*. 2003.

[130] O. Giraldo. *Nanotate - Guidelines*. July 2021. URL: `https://zenodo.org/record/5101941`.

[131] G. V. Glass. "Primary, Secondary, and Meta-Analysis of Research". In: *Educational researcher* (1976).

[132] A. Gómez-Pérez. “Ontology Evaluation”. In: *Handbook on Ontologies*. 2004.

[133] S. Gottschalk and E. Demidova. *EventKG*. Version 3.1. 2020. URL: https://doi.org/10.5281/zenodo.4720078.

[134] S. Gottschalk and E. Demidova. “EventKG - the Hub of Event Knowledge on the Web - and Biographical Timeline Generation”. In: *SWJ* (2019).

[135] S. Gottschalk and E. Demidova. “Eventkg: A Multilingual Event-centric Temporal Knowledge Graph”. In: *ESWC*. 2018.

[136] J. Gottschall. *The Storytelling Animal: How Stories Make Us Human*. 2012.

[137] P. Groth, A. Gibson, and J. Velterop. “The Anatomy of a Nano-publication”. In: *Information Services and Use* (2010).

[138] X. Gu and M. Krenn. “Forecasting high-impact research topics via machine learning on evolving knowledge graphs”. In: *Machine Learning: Science and Technology* (2025).

[139] S. Guan, X. Cheng, L. Bai, F. Zhang, Z. Li, Y. Zeng, X. Jin, and J. Guo. “What is Event Knowledge Graph: A Survey”. In: *IEEE Transactions on Knowledge and Data Engineering* (2022).

[140] S. Guan, X. Jin, J. Guo, Y. Wang, and X. Cheng. “NeuInfer: Knowledge Inference on N-ary Facts”. In: *ACL*. 2020.

[141] S. Guan, X. Jin, Y. Wang, and X. Cheng. “Link Prediction on N-ary Relational Data Based on Relatedness Evaluation”. In: *WWW*. 2019.

[142] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang. “REALM: Retrieval-Augmented Language Model Pre-Training”. In: *ICML*. 2020.

[143] T. Guyet. “Enhancing sequential pattern mining with time and reasoning”. PhD thesis. Université de Rennes 1, 2020.

[144] S. Harris, A. Seaborne, and E. Prud’hommeaux. *SPARQL 1.1 Query Language. W3C Recommendation (2013)*. 2013. URL: https://www.w3.org/TR/sparql11-query.

[145] O. Hartig. “Foundations of RDF and SPARQL (An Alternative Approach to Statement-Level Metadata in RDF)”. In: *Alberto Mendelzon International Workshop on Foundations of Data Management and the Web*. 2017.

[146] O. Hartig. “SQUIN: A Traversal Based Query Execution System for the Web of Linked Data”. In: *International Conference on Management of Data*. 2013.

[147] O. Hartig, C. Bizer, and J.-C. Freytag. “Executing SPARQL Queries over the Web of Linked Data”. In: *ISWC*. 2009.

[148] O. Hartig and J.-C. Freytag. “Foundations of Traversal Based Query Execution over Linked Data”. In: *ACM Conference on Hypertext and Social Media*. 2012.

[149] O. Hartig and M. T. Özsu. “Walking Without a Map: Ranking-Based Traversal for Querying Linked Data”. In: *ISWC*. 2016.

[150] O. Hartig and J. Pérez. “LDQL: A Query Language for the Web of Linked Data”. In: *JWS* (2016).

[151] O. Hartig and B. Thompson. “Foundations of an Alternative Approach to Reification in RDF”. In: *arXiv preprint arXiv:1406.3399* (2014).

[152] N. Heist, S. Hertling, D. Ringler, and H. Paulheim. “Knowledge Graphs on the Web - an Overview”. In: *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges* (2020).

[153] S. Hellmann, J. Lehmann, S. Auer, and M. Brümmer. “Integrating NLP using Linked Data”. In: *ISWC*. 2013.

[154] J. Henrich, R. Boyd, S. Bowles, C. Camerer, E. Fehr, H. Gintis, and R. McElreath. “In Search of Homo Economicus: Behavioral Experiments in 15 Small-Scale Societies”. In: *American Economic Review* (2001).

[155] D. Herman. “Storytelling and the Sciences of Mind: Cognitive Narratology, Discursive Psychology, and Narratives in Face-to-Face Interaction”. In: *Narrative* (2007).

[156] D. Hernández, A. Hogan, and M. Krötzsch. “Reifying RDF: What Works Well With Wikidata?” In: *International Workshop on Scalable Semantic Web Knowledge Base Systems @ISWC* (2015).

[157] J. E. T. Herrera, M. A. Casanova, B. P. Nunes, G. R. Lopes, and L. Leme. “DBpedia Profiler Tool: Profiling the Connectivity of Entity Pairs in DBpedia”. In: *International Workshop on Intelligent Exploration of Semantic Data*. 2016.

[158] J. E. T. Herrera. “On the Connectivity of Entity Pairs in Knowledge Bases”. PhD thesis. Pontifical Catholic University of Rio de Janeiro, 2017.

[159] H. Hlomani and D. Stacey. “Approaches, Methods, Metrics, Measures, and Subjectivity in Ontology Evaluation: A Survey”. In: *SWJ* (2014).

[160] A. Hogan. “The Semantic Web: Two Decades On”. In: *SWJ* (2020).

[161] A. Hogan et al. “Knowledge Graphs”. In: *ACM Computing Surveys* (2022).

[162] Z. Hu, V. Gutiérrez-Basulto, Z. Xiang, R. Li, and J. Z. Pan. “HyperFormer: Enhancing Entity and Relation Interaction for Hyper-Relational Knowledge Graph Completion”. In: *CIKM*. 2023.

[163] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al. “A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions”. In: *ACM Transactions on Information Systems* (2025).

[164] X. Huang, M. R. Orth, P. Barcelo, M. M. Bronstein, and I. I. Ceylan. “Link Prediction with Relational Hypergraphs”. In: *Transactions on Machine Learning Research* (2024).

[165] N. Hubert, P. Monnin, and H. Paulheim. “Beyond Transduction: A Survey on Inductive, Few Shot, and Zero Shot Link Prediction in Knowledge Graphs”. In: *arXiv preprint arXiv:2312.04997* (2023).

[166] P. Hühn, J. C. Meister, J. Pier, W. Schmid, and J. Schönert. *The living handbook of narratology*. 2009. URL: `http://www.lhn.uni-hamburg.de`.

[167] I. Hulpuş, N. Prangnawarat, and C. Hayes. “Path-Based Semantic Relatedness on Linked Data and Its Use to Word and Entity Disambiguation”. In: *ISWC*. 2015.

[168] E. Hyvönen, P. Leskinen, M. Tamper, H. Rantala, E. Ikkala, J. Tuominen, and K. Keravuori. “BiographySampo – Publishing and Enriching Biographies on the Semantic Web for Digital Humanities Research”. In: *ESWC*. 2019.

[169] A. Iglesias-Molina, K. Ahrabian, F. Ilievski, J. Pujara, and O. Corcho. “Comparison of Knowledge Graph Representations for Consumer Scenarios”. In: *ISWC*. 2023.

[170] R. Isele, J. Umbrich, C. Bizer, and A. Harth. “LDspider: An Open-source Crawling Framework for the Web of Linked Data”. In: *ISWC (Posters & Demos)*. 2010.

[171] K. Janowicz, A. Haller, S. J. Cox, D. Le Phuoc, and M. Lefrançois. “SOSA: A Lightweight Ontology for Sensors, Observations, Samples, and Actuators”. In: *JWS* (2019).

[172] K. Janowicz, P. Hitzler, B. Adams, D. Kolas, and C. Vardeman II. “Five Stars of Linked Data Vocabulary Use”. In: *SWJ* (2014).

[173] G. Jeh and J. Widom. “SimRank: A Measure of Structural-Context Similarity”. In: *KDD*. 2002.

[174] Z. Jia, S. Pramanik, R. Saha Roy, and G. Weikum. “Complex Temporal Question Answering on Knowledge Graphs”. In: *CIKM*. 2021.

[175] J. G. Jiménez, L. A. P. P. Leme, and M. A. Casanova. “CoEPinKB: Evaluating Path Search Strategies in Knowledge Bases”. In: *Journal of the Brazilian Computer Society* (2022).

[176] Q. Jin, R. Leaman, and Z. Lu. “PubMed and Beyond: Biomedical Literature Search in the Age of Artificial Intelligence”. In: *EBioMedicine* (2024).

[177] S. Jin, G. Spadaro, and D. Balliet. “Institutions and Cooperation: A Meta-Analysis of Structural Features in Social Dilemmas”. In: *Journal of Personality and Social Psychology* (2024).

[178] J. Kaddour, J. Harris, M. Mozes, H. Bradley, R. Raileanu, and R. McHardy. “Challenges and Applications of Large Language Models”. In: *arXiv preprint arXiv:2307.10169* (2023).

[179] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. “Dense Passage Retrieval for Open-Domain Question Answering”. In: *EMNLP*. 2020.

[180] V. Kattagoni and N. Singh. “IREvent2Story: A Novel Mediation Ontology and Narrative Generation”. In: *Text2Story @ECIR*. 2018.

[181] J. R. Katukuri, Y. Xie, V. V. Raghavan, and A. Gupta. “Hypotheses Generation as Supervised Link Discovery with Automated Class Labeling on Large-scale Biomedical Concept Networks”. In: *BioMed Central Genomics* (2012).

[182] T. Kawamura, S. Egami, K. Tamura, Y. Hokazono, T. Ugai, Y. Koyanagi, F. Nishino, S. Okajima, K. Murakami, K. Takamatsu, et al. “Report on the First Knowledge Graph Reasoning Challenge 2018 - Toward the eXplainable AI System”. In: *Joint International Semantic Technology Conference*. 2019.

[183] M. Kejriwal, J. Peng, H. Zhang, and P. Szekely. “Structured Event Entity Resolution in Humanitarian Domains”. In: *ISWC*. 2018.

[184] A. F. Khan, A. Bellandi, G. Benotto, F. Frontini, E. Giovannetti, and M. Reboul. “Leveraging a Narrative Ontology to Query a Literary Text”. In: *CMN*. 2016.

[185] R. P. Khandpur, T. Ji, S. Jan, G. Wang, C.-T. Lu, and N. Ramakrishnan. "Crowdsourcing Cybersecurity: Cyber Attack Detection using Social Media". In: *CIKM*. 2017.

[186] H. Khrouf and R. Troncy. "EventMedia: a LOD Dataset of Events Illustrated with Media". In: *SWJ* (2016).

[187] S. Kim, S. Y. Lee, Y. Gao, A. Antelmi, M. Polato, and K. Shin. "A Survey on Hypergraph Neural Networks: An In-Depth and Step-By-Step Guide". In: *KDD*. 2024.

[188] P. Kingsbury and M. Palmer. "PropBank: the Next Level of TreeBank". In: *Treebanks and Lexical Theories*. 2003.

[189] H. Kitano. "Nobel Turing Challenge: creating the engine for scientific discovery". In: *Nature Partner Journals Systems Biology and Applications* (2021).

[190] K. J. Kochut and M. Janik. "SPARQLeR: Extended Sparql for Semantic Association Discovery". In: *ESWC*. 2007.

[191] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. "Large Language Models are Zero-Shot Reasoners". In: *NeurIPS* (2022).

[192] R. Kowalski and M. Sergot. "A Logic-based Calculus of Events". In: *New Generation Computing* (1986).

[193] B. Krabina. "Building a Knowledge Graph for the History of Vienna with Semantic MediaWiki". In: *JWS* (2023).

[194] F. Krause, K. Kurniawan, E. Kiesling, H. Paulheim, and A. Polleres. "On the Representation of Dynamic BPMN Process Executions in Knowledge Graphs". In: *Iberoamerican Knowledge Graphs and Semantic Web Conference*. 2023.

[195] H. Kroll, D. Nagel, and W.-T. Balke. "Modeling Narrative Structures in Logical Overlays on Top of Knowledge Repositories". In: *International Conference on Conceptual Modeling*. 2020.

[196] W. H. Kruskal and W. A. Wallis. "Use of Ranks in One-Criterion Variance Analysis". In: *Journal of the American Statistical Association* (1952).

[197] B. Kybartas and R. Bidarra. "A Survey on Story Generation Techniques for Authoring Computational Narratives". In: *IEEE Transactions on Computational Intelligence and AI in Games* (2016).

[198] C. Lagoze and J. Hunter. "The ABC Ontology and Model". In: *Journal of Digital Information* (2002).

[199] D. Lahav, J. S. Falcon, B. Kuehl, S. Johnson, S. Parasa, N. Shomron, D. H. Chau, D. Yang, E. Horvitz, D. S. Weld, et al. “A Search Engine for Discovery of Scientific Challenges and Directions”. In: *AAAI*. 2022.

[200] K. F. Lawrence, M. M. Tuffield, M. O. Jewell, A. Prugel-Bennett, D. E. Millard, M. S. Nixon, N. R. Shadbolt, et al. “OntoMedia - Creating an Ontology for Marking Up the Contents of Heterogeneous Media”. In: *Ontology Patterns for the Semantic Web @ISWC*. 2005.

[201] F. Lecue. “On the Role of Knowledge Graphs in Explainable AI”. In: *SWJ* (2020).

[202] J. Lee, C. Chung, and J. J. Whang. “InGram: Inductive Knowledge Graph Embedding via Relation Graphs”. In: *ICML*. 2023.

[203] K. Lee, M.-W. Chang, and K. Toutanova. “Latent Retrieval for Weakly Supervised Open Domain Question Answering”. In: *ACL*. 2019.

[204] K. Leetaru and P. A. Schrodt. “GDELT - Global Data on Events, Location and Tone, 1979-2012”. In: *International Studies Association Annual Convention*. 2013.

[205] T. I. Leung, T. Kuhn, and M. Dumontier. “Representing Physician Suicide Claims as Nanopublications: Proof-of-Concept Study Creating Claim Networks”. In: *Journal of Medical Internet Research x Med* (2022).

[206] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”. In: *NeurIPS* (2020).

[207] Y. Li, H. Chen, X. Sun, Z. Sun, L. Li, L. Cui, P. S. Yu, and G. Xu. “Hyperbolic Hypergraphs for Sequential Recommendation”. In: *CIKM*. 2021.

[208] Z. Li, C. Wang, X. Wang, Z. Chen, and J. Li. “HJE: Joint Convolutional Representation Learning for Knowledge Hypergraph Completion”. In: *IEEE Transactions on Knowledge and Data Engineering* (2024).

[209] Z. Li, X. Jin, S. Guan, W. Li, J. Guo, Y. Wang, and X. Cheng. “Search from History and Reason for Future: Two-stage Reasoning on Temporal Knowledge Graphs”. In: *ACL-IJCNLP (Long Papers)*. 2021.

[210] X. Liang, G. Si, J. Li, P. Tian, Z. An, and F. Zhou. “A survey of inductive knowledge graph completion”. In: *Neural Computing and Applications* (2024).

[211] H. Lim, J. Y. Cho, T. Kim, J. Park, H. Shin, S. Choi, S. Park, K. Lee, J. Kim, M. Lee, et al. “Co-Creating Question-and-Answer Style Articles with Large Language Models for Research Promotion”. In: *Conference on Designing Interactive Systems*. 2024.

[212] P. Lisena, D. Schwabe, M. van Erp, R. Troncy, W. Tullett, I. Leemans, L. Marx, and S. C. Ehrich. “Capturing the Semantics of Smell: The Odeuropa Data Model for Olfactory Heritage Information”. In: *ESWC*. 2022.

[213] Q. Liu, G. Cheng, K. Gunaratna, and Y. Qu. “Entity Summarization: State of the Art and Future Challenges”. In: *JWS* (2021).

[214] X. Liu, K. Li, M. Zhou, and Z. Xiong. “Collective Semantic Role Labeling for Tweets with Clustering”. In: *IJCAI*. 2011.

[215] X. Liu, J. Jin, Q. Wang, T. Ruan, Y. Zhou, D. Gao, and Y. Yin. “PatientEG Dataset: Bringing Event Graph Model with Temporal Relations to Electronic Medical Records”. In: *arXiv preprint arXiv:1812.09905* (2018).

[216] Y. Liu, Q. Yao, and Y. Li. “Generalizing Tensor Decomposition for N-ary Relational Knowledge Bases”. In: *WWW*. 2020.

[217] Y. Liu, Q. Yao, and Y. Li. “Role-Aware Modeling for N-ary Relational Knowledge Bases”. In: *WWW*. 2021.

[218] Y. Liu, M. Hildebrandt, M. Joblin, M. Ringsquandl, R. Raissouni, and V. Tresp. “Neural Multi-Hop Reasoning With Logical Rules on Biomedical Knowledge Graphs”. In: *ESWC*. 2021.

[219] W.-Y. Loh. “Classification and Regression Trees”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (2011).

[220] S. Lohmann, P. Heim, T. Stegemann, and J. Ziegler. “The RelFinder User Interface: Interactive Exploration of Relationships between Objects of Interest”. In: *International Conference on Intelligent User Interfaces*. 2010.

[221] V. Lombardo, R. Damiano, and A. Pizzo. “Drammar: A Comprehensive Ontological Resource on Drama”. In: *ISWC*. 2018.

[222] Y. Lu, D. Yang, P. Wang, P. Rosso, and P. Cudre-Mauroux. “Schema-Aware Hyper-Relational Knowledge Graph Embeddings for Link Prediction”. In: *IEEE Transactions on Knowledge and Data Engineering* (2023).

[223] H. Luo, E. Haihong, Y. Yang, Y. Guo, M. Sun, T. Yao, Z. Tang, K. Wan, M. Song, and W. Lin. "HAHE: Hierarchical Attention for Hyper-Relational Knowledge Graphs in Global and Local Level". In: *ACL (Long Papers)*. 2023.

[224] R. O. Lutkenhaus, J. Jansz, and M. P. Bouman. "Mapping the Dutch vaccination debate on Twitter: Identifying communities, narratives, and interactions". In: *Vaccine: X* (2019).

[225] S. J. Lynden, I. Kojima, A. Matono, A. Nakamura, and M. Yui. "A Hybrid Approach to Linked Data Query Processing with Time Constraints". In: *Workshop on Linked Data on the Web @WWW*. 2013.

[226] I. Magnusson and S. Friedman. "Extracting Fine-Grained Knowledge Graphs of Scientific Claims: Dataset and Transformer-Based Results". In: *EMNLP*. 2021.

[227] H. B. Mann and D. R. Whitney. "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other". In: *The Annals of Mathematical Statistics* (1947).

[228] W. C. Mann and S. A. Thompson. *Rhetorical structure theory: A theory of text organization*. 1987.

[229] F. Manola and E. Miller. "RDF Primer: W3c Recommendation". In: *Decision Support Systems* (2004).

[230] T. Marcoux and N. Agarwal. "Narrative Trends of COVID-19 Misinformation". In: *Text2Story @ECIR*. 2021.

[231] E. Markowitz, K. Balasubramanian, M. Mirtaheri, M. Annavaram, A. Galstyan, and G. Ver Steeg. "StATIK: Structure and Text for Inductive Knowledge Graph Completion". In: *Findings of NAACL*. 2022.

[232] V. Mastoras, A. Vassiliades, M. Rousi, S. Diplaris, T. Mavropoulos, I. Gialampoukidis, S. Vrochidis, and I. Kompatsiaris. "Towards a Framework for Seismic Data". In: *Iberoamerican Knowledge Graphs and Semantic Web Conference*. 2023.

[233] J. P. McCrae, J. Bosque-Gil, J. Gracia, P. Buitelaar, and P. Cimiano. "The OntoLex-Lemon Model: Development and Applications". In: *Electronic Lexicography in the 21st Century Conference*. 2017.

[234] A. L. Mederos, D. García-Duarte, D. G. Lio, Y. Hidalgo-Delgado, and J. A. S. Ruíz. "An Ontological Model for the Failure Detection in Power Electric Systems". In: *Iberoamerican Knowledge Graphs and Semantic Web Conference*. 2020.

[235] C. Meghini, V. Bartalesi, and D. Metilli. “Representing Narratives in Digital Libraries: The Narrative Ontology”. In: *SWJ* (2021).

[236] C. Meilicke, M. W. Chekol, D. Ruffinelli, and H. Stuckenschmidt. “Anytime Bottom-Up Rule Learning for Knowledge Graph Completion”. In: *IJCAI*. 2019.

[237] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. “DBpedia Spotlight: Shedding Light on the Web of Documents”. In: *International Conference on Semantic Systems*. 2011.

[238] A. Meroño-Peñuela, A. Ashkpour, M. Van Erp, K. Mandemakers, L. Breure, A. Scharnhorst, S. Schlobach, and F. Van Harmelen. “Semantic Technologies for Historical Research: A Survey”. In: *SWJ* (2015).

[239] A. Meroño-Peñuela and R. Hoekstra. “What Is Linked Historical Data?” In: *EKAW*. 2014.

[240] B. Miller, A. Lieto, R. Ronfard, S. Ware, and M. Finlayson. “7th CMN”. In: *CMN*. 2016.

[241] J. L. Moore, F. Steinke, and V. Tresp. “A Novel Metric for Information Retrieval in Semantic Networks”. In: *ESWC*. 2011.

[242] P. Mulholland, T. Collins, and Z. Zdrahal. “Story Fountain: Intelligent Support for Story Research and Exploration”. In: *International Conference on Intelligent User Interfaces*. 2004.

[243] P. Mulholland, A. Wolff, and T. Collins. “Curate and Storyspace: An Ontology and Web-Based Environment for Describing Curatorial Narratives”. In: *ESWC*. 2012.

[244] P. Mulholland, A. Wolff, E. Kilfeather, and E. McCarthy. “Using Event Spaces, Setting and Theme to Assist the Interpretation and Development of Museum Stories”. In: *EKAW*. 2014.

[245] M. Nagarajan, A. D. Wilkins, B. J. Bachman, I. B. Novikov, S. Bao, P. J. Haas, M. E. Terrón-Díaz, S. Bhatia, A. K. Adikesavan, J. J. Labrie, et al. “Predicting Future Scientific Discoveries Based on a Networked Analysis of the Past Literature”. In: *KDD*. 2015.

[246] A. Nakasone and M. Ishizuka. “Storytelling Ontology Model Using RST”. In: *International Conference on Intelligent Agent Technology*. 2006.

[247] K.-H. Nguyen, X. Tannier, O. Ferret, and R. Besançon. “A Dataset for Open Event Extraction in English”. In: *Workshop on Language Resources for Linguistic Creativity*. 2016.

[248] V. Nguyen, O. Bodenreider, and A. Sheth. “Don’t Like RDF Reification? Making Statements about Statements Using Singleton Property”. In: *WWW*. 2014.

[249] J. Noordegraaf, M. Van Erp, R. Zijdeman, M. Raat, T. Van Oort, I. Zandhuis, T. Vermaut, H. Mol, N. Van der Sijs, K. Doreleijers, et al. “Semantic Deep Mapping in the Amsterdam Time Machine: Viewing Late 19th- and Early 20th-Century Theatre and Cinema Culture Through the Lens of Language Use and Socio-Economic Status”. In: *Workshop on Research and Education in Urban History in the Age of Digital Libraries*. 2019.

[250] A. Nowak, P. Lukowicz, and P. Horodecki. “Assessing Artificial Intelligence for Humanity: Will AI be the Our Biggest Ever Advance ? or the Biggest Threat [Opinion]”. In: *IEEE Technology and Society Magazine* (2018).

[251] E. T. O’Neill. “Frbr: Functional Requirements for Bibliographic Records”. In: *Library Resources & Technical Services* (2011).

[252] D. Oldman and D. Tanase. “Reshaping the Knowledge Graph by Connecting Researchers, Data and Practices in ResearchSpace”. In: *ISWC*. 2018.

[253] Ontotext. *What Is RDF-star?* 2025. URL: `https://www.ontotext.com/knowledgehub/fundamentals/what-is-rdf-star/`.

[254] OpenAI. *Introducing ChatGPT*. 2022. URL: `https://openai.com/index/chatgpt/`.

[255] M. J. Page, D. Moher, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, et al. “PRISMA 2020 Explanation and Elaboration: Updated Guidance and Exemplars for Reporting Systematic Reviews”. In: *BMJ* (2021).

[256] A. Pak, P. Paroubek, et al. “Twitter as a Corpus for Sentiment Analysis and Opinion Mining”. In: *Workshop on Language Resources for Linguistic Creativity*. 2010.

[257] V. Pankratius, J. Li, M. Gowanlock, D. M. Blair, C. Rude, T. Herring, F. Lind, P. J. Erickson, and C. Lonsdale. “Computer-Aided Discovery: Toward Scientific Insight Generation with Machine Support”. In: *IEEE Intelligent Systems* (2016).

[258] F. Pannach, C. Sporleder, W. May, A. Krishnan, and A. Sewchurran. “Of Lions and Yakshis: Ontology-based Narrative Structure Modelling for Culturally Diverse Folktales”. In: *SWJ* (2021).

[259] F. Peinado, P. Gervás, and B. Díaz-Agudo. “A Description Logic Ontology for Fairy Tale Generation”. In: *Workshop on Language Resources for Linguistic Creativity*. 2004.

[260] T. Pellissier Tanon, G. Weikum, and F. Suchanek. “YAGO 4: A Reasonable Knowledge Base”. In: *ESWC*. 2020.

[261] L. Penev, T. Kuhn, R. L. Pyle, D. Mietchen, S. Demirov, I. Boyadzhieva, and T. Georgiev. “Nanopublications for Biodiversity Go Live”. In: *Biodiversity Information Science and Standards* (2023).

[262] C. Peng, F. Xia, M. Naseriparsa, and F. Osborne. “Knowledge graphs: Opportunities and Challenges”. In: *AI Review* (2023).

[263] J. Pérez, M. Arenas, and C. Gutierrez. “nSPARQL: A Navigational Language for RDF”. In: *JWS* (2010).

[264] F. Petroni, P. Lewis, A. Piktus, T. Rocktäschel, Y. Wu, A. H. Miller, and S. Riedel. “How Context Affects Language Models’ Factual Predictions”. In: *Automated Knowledge Base Construction*. 2020.

[265] F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. “Language Models as Knowledge Bases?” In: *EMNLP-IJCNLP*. 2019.

[266] G. Pirrò. “Explaining and Suggesting Relatedness in Knowledge Graphs”. In: *ISWC*. 2015.

[267] R. Piryani, N. Aussenac-Gilles, and N. J. Hernandez. “Comprehensive Survey on Ontologies about Event”. In: *Semantic Methods for Events and Stories @ESWC*. 2023.

[268] R. Porzel. “On Formalizing Narratives”. In: *Joint Ontology Workshops*. 2021.

[269] V. Propp. *Morphology of the Folktale*. 1968.

[270] P. Proutskova, D. Wolff, G. Fazekas, K. Frieler, F. Höger, O. Velichkina, G. Solis, T. Weyde, M. Pfleiderer, H. C. Crayencour, et al. “The Jazz Ontology: A Semantic Model and Large-scale RDF Repositories for Jazz”. In: *JWS* (2022).

[271] J. Raad and C. Cruz. “A Survey on Ontology Evaluation Methods”. In: *International Conference on Knowledge Engineering and Ontology Development*. 2015.

[272] Y. Raimond and S. Abdallah. *The Event Ontology*. 2007. URL: `https://lov.linkeddata.es/dataset/lov/vocabs/event`.

[273] G. Razis and I. Anagnostopoulos. “Semantifying Twitter: the influenceTracker ontology”. In: *International Workshop on Semantic and Social Media Adaptation and Personalization*. 2014.

[274] S. Razniewski, A. Yates, N. Kassner, and G. Weikum. “Language Models As or For Knowledge Bases”. In: *Deep Learning for Knowledge Graphs*. 2021.

[275] Y. Rebboud, P. Lisena, and R. Troncy. “Beyond Causality: Representing Event Relations in Knowledge Graphs”. In: *EKAW*. 2022.

[276] J. L. Reutter, A. Soto, and D. Vrgoč. “Recursion in SPARQL”. In: *ISWC*. 2015.

[277] M. O. Riedl. “Computational Narrative Intelligence: A Human-Centered Goal for Artificial Intelligence”. In: *arXiv preprint arXiv:1602.06484* (2016).

[278] A. Roberts, C. Raffel, and N. Shazeer. “How Much Knowledge Can You Pack Into the Parameters of a Language Model?” In: *EMNLP*. 2020.

[279] M. Rospocher, M. van Erp, P. Vossen, A. Fokkens, I. Aldabe, G. Rigau, A. Soroa, T. Ploeger, and T. Bogaard. “Building Event-centric Knowledge Graphs from News”. In: *JWS* (2016).

[280] P. Rosso, D. Yang, and P. Cudré-Mauroux. “Beyond Triplets: Hyper-Relational Knowledge Graph Embedding for Link Prediction”. In: *WWW*. 2020.

[281] H. Saif, Y. He, and H. Alani. “Semantic Sentiment Analysis of Twitter”. In: *ISWC*. 2012.

[282] A. A. Salatino, Y. Bu, Y. Ding, Á. Horvát, Y. Huang, M. Liu, P. Manghi, A. Mannocci, F. Osborne, D. Romero, et al. “3rd International Workshop on Scientific Knowledge Representation, Discovery, and Assessment (Sci-K 2023)”. In: *Companion of the WWW*. 2023.

[283] C. R. M. A. Santagiustina and M. Warglien. “The Architecture of Partisan Debates: The Online Controversy on the No-deal Brexit”. In: *PLoS One* (2022).

[284] R. Sapkota, S. Raza, and M. Karkee. “Comprehensive Analysis of Transparency and Accessibility of ChatGPT, DeepSeek, And other SoTA Large Language Models”. In: *arXiv preprint arXiv:2502.18505* (2025).

[285] J. Sardina, J. D. Kelleher, and D. O’Sullivan. “A Survey on Knowledge Graph Structure and Knowledge Graph Embeddings”. In: *arXiv preprint arXiv:2412.10092* (2024).

[286] S. S. Sawilowsky. “New Effect Size Rules of Thumb”. In: *Journal of Modern Applied Statistical Methods* (2009).

[287] S. Schaffert, C. Bauer, T. Kurz, F. Dorschel, D. Glachs, and M. Fernandez. “The Linked Media Framework: Integrating and interlinking enterprise media content and data”. In: *International Conference on Semantic Systems*. 2012.

[288] R. C. Schank and R. P. Abelson. *Scripts, plans, and knowledge*. 1975.

[289] *Schema.org*. 2024. URL: `https://schema.org`.

[290] A. Scherp, T. Franz, C. Saathoff, and S. Staab. “F – A Model of Events based on the Foundational Ontology DOLCE+DnS Ultralite”. In: *K-CAP*. 2009.

[291] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling. “Modeling Relational Data with Graph Convolutional Networks”. In: *ESWC*. 2018.

[292] S. Schouten, V. De Boer, L. Petram, and M. Van Erp. “The Wind in Our Sails: Developing a Reusable and Maintainable Dutch Maritime History Knowledge Graph”. In: *K-CAP*. 2021.

[293] Z. Shao, R. Zhao, S. Yuan, M. Ding, and Y. Wang. “Tracing the evolution of AI in the past decade and forecasting the emerging trends”. In: *Expert Systems with Applications* (2022).

[294] R. Shaw, R. Troncy, and L. Hardman. “Lode: Linking Open Descriptions of Events”. In: *Asian Semantic Web Conference*. 2009.

[295] H. Shen, C.-Y. Huang, T. Wu, and T.-H. K. Huang. “ConvXAI: Delivering Heterogeneous AI Explanations via Conversations to Support Human-AI Scientific Writing”. In: *Companion of the Conference on Computer Supported Cooperative Work and Social Computing*. 2023.

[296] C. Shimizu, P. Hitzler, Q. Hirt, D. Rehberger, S. G. Estrecha, C. Foley, A. M. Sheill, W. Hawthorne, J. Mixter, E. Watrall, et al. “The Enslaved Ontology: Peoples of the Historic Slave Trade”. In: *JWS* (2020).

[297] C. Si, D. Yang, and T. Hashimoto. “Can LLMs Generate Novel Research Ideas? A Large-Scale Human Study with 100+ NLP Researchers”. In: *ICLR*. 2025.

[298] J. Singh, W. Nejdl, and A. Anand. “History by Diversity: Helping Historians search News Archives”. In: *Conference on Human Information Interaction and Retrieval*. 2016.

[299] L. Sinikallio, S. Drobac, M. Tamper, R. Leal, M. Koho, J. Tuominen, M. L. Mela, E. Hyvönen, et al. “Plenary Debates of the Parliament of Finland as Linked Open Data and in Parla-CLARIN Markup”. In: *LDK*. 2021.

[300] S. A. Sloman. “The Empirical Case for Two Systems of Reasoning”. In: *Psychological Bulletin* (1996).

[301] A. Søgaard, B. Plank, and H. Alonso. “Using Frame Semantics for Knowledge Extraction from Twitter”. In: *AAAI*. 2015.

[302] D. N. Sosa, A. Derry, M. Guo, E. Wei, C. Brinton, and R. B. Altman. “A Literature-Based Knowledge Graph Embedding Method for Identifying Drug Repurposing Opportunities in Rare Diseases”. In: *Pacific Symposium on Biocomputing*. 2019.

[303] T. Souza Costa, S. Gottschalk, and E. Demidova. “Event-QA: A Dataset for Event-Centric Question Answering over Knowledge Graphs”. In: *CIKM*. 2020.

[304] G. Spadaro, C. Graf, S. Jin, S. Arai, Y. Inoue, E. Lieberman, M. I. Rinderu, M. Yuan, C. J. Van Lissa, and D. Balliet. “Cross-Cultural Variation in Cooperation: A Meta-Analysis”. In: *Journal of Personality and Social Psychology* (2022).

[305] G. Spadaro, S. Jin, and D. Balliet. “Gender differences in cooperation across 20 societies: a meta-analysis”. In: *Philosophical Transactions of the Royal Society B* (2023).

[306] G. Spadaro, I. Tiddi, S. Columbus, S. Jin, A. Ten Teije, C. Team, and D. Balliet. “The Cooperation Databank: Machine-Readable Science Accelerates Research Synthesis”. In: *Perspectives on Psychological Science* (2022).

[307] L. Spillner, C. R. M. A. Santagiustina, T. Mildner, and R. Porzel. “Towards Conflictual Narrative Mechanics”. In: *Workshop on Semantic Techniques for Narrative-based Understanding @IJCAI*. 2022.

[308] P. Srinivasan. “Text mining: Generating hypotheses from MEDLINE”. In: *Journal of the American Society for Information Science and Technology* (2004).

[309] S. Stadtmüller, S. Speiser, A. Harth, and R. Studer. “Data-Fu: A Language and an Interpreter for Interaction with Read/Write Linked Data”. In: *WWW*. 2013.

[310] L. Steels. “Foundations for Meaning and Understanding in Human-centric AI”. In: *Venice: Venice International University* (2022).

[311] L. Steels, L. Verheyen, and R. van Trijp. “An Experiment in Measuring Understanding”. In: *HHAI*. 2022.

[312] M. Stocker, A. Oelen, M. Y. Jaradeh, M. Haris, O. A. Oghli, G. Heidari, H. Hussein, A.-L. Lorenz, S. Kabenamualu, K. E. Farfar, et al. “FAIR scientific information with the Open Research Knowledge Graph”. In: *FAIR Connect* (2023).

[313] L. Stork, I. Tiddi, R. Spijker, and A. ten Teije. “Explainable Drug Repurposing in Context via Deep Reinforcement Learning”. In: *ESWC*. 2023.

[314] L. Stork, R. L. Zijdeman, I. Tiddi, and A. ten Teije. “Enabling Social Demography Research Using Semantic Technologies”. In: *ESWC*. 2024.

[315] R. Stühmer, D. Anicic, S. Sen, J. Ma, K.-U. Schmidt, and N. Stojanovic. “Lifting Events in RDF from Interactions with Annotated Web Pages”. In: *ISWC*. 2009.

[316] F. M. Suchanek. “The Need to Move beyond Triples”. In: *Text2Story @ECIR*. 2020.

[317] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. Ni, H.-Y. Shum, and J. Guo. “Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph”. In: *ICLR*. 2024.

[318] D. R. Swanson and N. R. Smalheiser. “An interactive system for finding complementary literatures: a stimulus to scientific discovery”. In: *AI* (1997).

[319] I. Swartjes and M. Theune. “A Fabula Model for Emergent Narrative”. In: *International Conference on Technologies for Interactive Digital Storytelling and Entertainment*. 2006.

[320] R. Taelman, J. Van Herwegen, M. Vander Sande, and R. Verborgh. “Comunica: A Modular SPARQL Query Engine for the Web”. In: *ISWC*. 2018.

[321] R. Taelman and R. Verborgh. “Link Traversal Query Processing Over Decentralized Environments with Structural Assumptions”. In: *ISWC*. 2023.

[322] C. Tao, H. R. Solbrig, D. K. Sharma, W.-Q. Wei, G. K. Savova, and C. G. Chute. “Time-Oriented Question Answering from Clinical Narratives Using Semantic-Web Techniques”. In: *ISWC*. 2010.

[323] L. Tavoschi, F. Quattrone, E. D'Andrea, P. Ducange, M. Vabanesi, F. Marcelloni, and P. L. Lopalco. "Twitter as a sentinel tool to monitor public opinion on vaccination: an opinion mining analysis from September 2016 to August 2017 in Italy". In: *Human Vaccines & Immunotherapeutics* (2020).

[324] K. Teru, E. Denis, and W. Hamilton. "Inductive Relation Prediction by Subgraph Reasoning". In: *ICML*. 2020.

[325] I. Tiddi, M. d'Aquin, and E. Motta. "Learning to Assess Linked Data Relationships Using Genetic Programming". In: *ISWC*. 2016.

[326] F. Tomasi, M. Daquino, L. Giagnolini, et al. "ARTchives: a Linked Open Data Native Catalogue of Art Historians' Archives". In: *International Workshop on Linked Archives*. 2021.

[327] S. Tong, K. Mao, Z. Huang, Y. Zhao, and K. Peng. "Automating Psychological Hypothesis Generation with AI: Large Language Models Meet Causal Graph". In: *Humanities and Social Sciences Communications* (2024).

[328] T. Trabasso, P. Van den Broek, and S. Y. Suh. "Logical Necessity and Transitivity of Causal Relations in Stories". In: *Discourse Processes* (1989).

[329] R. Troncy, G. Rizzo, A. Jameson, O. Corcho, J. Plu, E. Palumbo, J. C. B. Hermida, A. Spirescu, K.-D. Kuhn, C. Barbu, et al. "3cixty: Building Comprehensive Knowledge Bases for City Exploration". In: *JWS* (2017).

[330] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, and G. Bouchard. "Complex Embeddings for Simple Link Prediction". In: *ICML*. 2016.

[331] M.-H. Tsou, C.-T. Jung, C. Allen, J.-A. Yang, J.-M. Gawron, B. H. Spitzberg, and S. Han. "Social media analytics and research test-bed (SMART dashboard)". In: *International Conference on Social Media & Society*. 2015.

[332] K. Tu, P. Cui, X. Wang, F. Wang, and W. Zhu. "Structural Deep Embedding for Hyper-Networks". In: *AAAI*. 2018.

[333] A. Tumasjan, T. Sprenger, P. Sandner, and I. Welpe. "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment". In: *AAAI Conference on Web and Social Media*. 2010.

[334] J. Umbrich, A. Hogan, A. Polleres, and S. Decker. "Improving the Recall of Live Linked Data Querying through Reasoning". In: *International Conference on Web Reasoning and Rule Systems*. 2012.

[335] J. Umbrich, A. Hogan, A. Polleres, and S. Decker. “Link Traversal Querying for a Diverse Web of Data”. In: *SWJ* (2015).

[336] C. Van den Akker, L. Aroyo, A. Cybulska, M. Van Erp, P. Gorgels, L. Hollink, C. Jager, S. Legene, L. van der Meij, J. Oomen, et al. “Historical Event-based Access to Museum Collections”. In: *Applied AI* (2010).

[337] R. van der Weerdt, V. de Boer, L. Daniele, R. Siebes, and F. van Harmelen. “Representation Learning on IoT Knowledge Graphs”. In: *Conference on Metadata and Semantics Research*. 2024.

[338] W. R. Van Hage, V. Malaisé, R. Segers, L. Hollink, and G. Schreiber. “Design and Use of the Simple Event Model (SEM)”. In: *JWS* (2011).

[339] U. Varadarajan and B. Dutta. “Models for Narrative Information: a Study”. In: *arXiv preprint arXiv:2110.02084* (2021).

[340] A. S. M. Venigalla, S. Chimalakonda, and D. Vagavolu. “Mood of India During Covid-19 - An Interactive Web Portal Based on Emotion Analysis of Twitter Data”. In: *Companion of the Conference on Computer Supported Cooperative Work and Social Computing*. 2020.

[341] Ó. Vilarroya. “Somos lo que nos contamos”. In: *Cómo los relatos construyen el mundo en que vivimos., Editorial Ariel, Barcelona* (2019).

[342] N. Voskarides, E. Meij, S. Sauer, and M. de Rijke. “News Article Retrieval in Context for Event-centric Narrative Creation”. In: *International Conference on Theory of Information Retrieval*. 2021.

[343] D. Vrandečić. “Ontology Evaluation”. In: *Handbook on Ontologies*. 2009.

[344] D. Vrandečić and M. Krötzsch. “Wikidata: A Free Collaborative Knowledge Base”. In: *Communications of the ACM* (2014).

[345] C. Wang, X. Wang, Z. Li, Z. Chen, and J. Li. “HyConvE: A Novel Embedding Model for Knowledge Hypergraph Link Prediction with Convolutional Neural Networks”. In: *WWW*. 2023.

[346] L. Wang, W. Zhao, Z. Wei, and J. Liu. “SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models”. In: *ACL (Long Papers)*. 2022.

[347] P. Wang, J. Chen, L. Su, and Z. Wang. “N-ary Relation Prediction based on Knowledge Graphs with Important Entity Detection”. In: *Expert Systems with Applications* (2023).

[348] Q. Wang, H. Wang, Y. Lyu, and Y. Zhu. “Link Prediction on N-ary Relational Facts: A Graph-based Approach”. In: *Findings of ACL*. 2021.

[349] X.-j. Wang, S. Mamadgi, A. Thekdi, A. Kelliher, and H. Sundaram. "Eventory – An Event Based Media Repository". In: *International Conference on Semantic Computing*. 2007.

[350] Y. Wang, J. He, and H. Wang. "RHKH: Relational Hypergraph Neural Network for Link Prediction on N-ary Knowledge Hypergraph". In: *International Conference on Multimedia*. 2024.

[351] C.-H. Wei, A. Allot, P.-T. Lai, R. Leaman, S. Tian, L. Luo, Q. Jin, Z. Wang, Q. Chen, and Z. Lu. "PubTator 3.0: an AI-powered literature resource for unlocking biomedical knowledge". In: *Nucleic Acids Research* (2024).

[352] J. Wei, S. Guan, X. Jin, J. Guo, and X. Cheng. "Few-shot Link Prediction on Hyper-relational Facts". In: *LREC-COLING*. 2024.

[353] J. Wei, S. Guan, X. Jin, J. Guo, and X. Cheng. "Inductive Link Prediction in N-ary Knowledge Graphs". In: *COLING*. 2025.

[354] E. W. Weisstein. *Bonferroni Correction*. 2004. URL: `https://mathworld.wolfram.com/BonferroniCorrection.html`.

[355] J. Wen, J. Li, Y. Mao, S. Chen, and R. Zhang. "On the Representation and Embedding of Knowledge Bases Beyond Binary Relations". In: *IJCAI*. 2016.

[356] U. Westermann and R. Jain. "Toward a Common Event Model for Multimedia Applications". In: *IEEE Multimedia* (2007).

[357] S. Whitfield and M. A. Hofmann. "Elicit: AI literature review research assistant". In: *Public Services Quarterly* (2023).

[358] T. Willaert, P. Van Eecke, K. Beuls, and L. Steels. "Building Social Media Observatories for Monitoring Online Opinion Dynamics". In: *Social Media + Society* (2020).

[359] J. M. Williamson. *Teaching to Individual Differences in Science and Engineering Librarianship: Adapting Library Instruction to Learning Styles and Personality Characteristics*. 2017.

[360] I. H. Witten and D. N. Milne. "An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links". In: *Workshop on Wikipedia and AI: an Evolving Synergy @AAAI*. 2008.

[361] J. Wu, X. Zhu, C. Zhang, and Z. Hu. "Event-centric Tourism Knowledge Graph — A Case Study of Hainan". In: *International Conference on Knowledge Science, Engineering and Management*. 2020.

[362] R. Xie, Z. Liu, J. Jia, H. Luan, and M. Sun. "Representation Learning of Knowledge Graphs with Entity Descriptions". In: *AAAI*. 2016.

[363] B. Xiong, M. Nayyeri, D. Daza, and M. Cochez. “Reasoning beyond Triples: Recent Advances in Knowledge Graph Embeddings”. In: *CIKM*. 2023.

[364] B. Xiong, M. Nayyeri, L. Luo, Z. Wang, S. Pan, and S. Staab. “NestE: Modeling Nested Relational Structures for Knowledge Graph Reasoning”. In: *AAAI*. 2024.

[365] B. Xiong, M. Nayyeri, S. Pan, and S. Staab. “Shrinking Embeddings for Hyper-Relational Knowledge Graphs”. In: *ACL (Long Papers)*. 2023.

[366] N. Yadati. “Neural Message Passing for Multi-Relational Ordered and Recursive Hypergraphs”. In: *NeurIPS* (2020).

[367] N. Yadati, M. Nimishakavi, P. Yadav, V. Nitin, A. Louis, and P. Talukdar. “HyperGCN: A New Method of Training Graph Convolutional Networks on Hypergraphs”. In: *NeurIPS* (2019).

[368] N. Yadati, V. Nitin, M. Nimishakavi, P. Yadav, A. Louis, and P. Talukdar. “NHP: Neural Hypergraph Link Prediction”. In: *CIKM*. 2020.

[369] S. Yan, Z. Zhang, X. Sun, G. Xu, L. Jin, and S. Li. “HYPER2: Hyperbolic embedding for hyper-relational link prediction”. In: *Neurocomputing* (2022).

[370] B. Yang, S. W.-t. Yih, X. He, J. Gao, and L. Deng. “Embedding Entities and Relations for Learning and Inference in Knowledge Bases”. In: *ICLR*. 2015.

[371] Y. Yang, X. Li, Y. Guan, H. Wang, C. Kong, and J. Jiang. “LHP: Logical Hypergraph Link Prediction”. In: *Expert Systems with Applications* (2023).

[372] J.-h. Yeh. “StoryTeller: An Event-based Story Ontology Composition System for Biographical History”. In: *International Conference on Applied System Innovation*. 2017.

[373] D. Yu and Y. Yang. “Improving Hyper-Relational Knowledge Graph Completion”. In: *arXiv preprint arXiv:2104.08167* (2021).

[374] M. Yuan, G. Spadaro, S. Jin, J. Wu, Y. Kou, P. A. Van Lange, and D. Balliet. “Did Cooperation Among Strangers Decline in the United States? A Cross-Temporal Meta-Analysis of Social Dilemmas (1956–2017)”. In: *Psychological Bulletin* (2022).

[375] M. Zárate, P. Rosales, G. Braun, M. Lewis, P. R. Fillottrani, and C. Delrieux. “OceanGraph: Some Initial Steps Toward a Oceanographic Knowledge Graph”. In: *Iberoamerican Knowledge Graphs and Semantic Web Conference*. 2019.

[376] H. Zauner, B. Linse, T. Furche, and F. Bry. “A RPL through RDF: Expressive Navigation in RDF Graphs”. In: *International Conference on Web Reasoning and Rule Systems*. 2010.

[377] R. Zhang, J. Li, J. Mei, and Y. Mao. “Scalable Instance Reconstruction in Knowledge Bases via Relatedness Affiliated Embedding”. In: *WWW*. 2018.

[378] R. Zhang, Y. Zou, and J. Ma. “Hyper-SAGNN: a self-attention based graph neural network for hypergraphs”. In: *ICLR*. 2019.

[379] W. Zhang, J. Chen, J. Li, Z. Xu, J. Z. Pan, and H. Chen. “Knowledge Graph Reasoning with Logics and Embeddings: Survey and Perspective”. In: *International Conference on Knowledge Graph*. 2024.

[380] Y. Zhang and Q. Yao. “Knowledge Graph Reasoning with Relational Digraph”. In: *WWW*. 2022.

[381] Y. Zhang, Z. Zhou, Q. Yao, X. Chu, and B. Han. “AdaProp: Learning Adaptive Propagation for Graph Neural Network based Knowledge Graph Reasoning”. In: *KDD*. 2023.

[382] Y. Zhang, B. Bevilacqua, M. Galkin, and B. Ribeiro. “TRIX: A More Expressive Model for Zero-shot Domain Transfer in Knowledge Graphs”. In: *Learning on Graphs Conference*. 2025.

[383] C. Zheng, J. Feng, Z. Fu, Y. Cai, Q. Li, and T. Wang. “Multimodal Relation Extraction with Efficient Graph Alignment”. In: *International Conference on Multimedia*. 2021.

[384] X. Zhou, B. Hui, I. Zeira, H. Wu, and L. Tian. “Dynamic relation learning for link prediction in knowledge hypergraphs”. In: *Applied Intelligence* (2023).

[385] Z. Zhu, X. Yuan, M. Galkin, L.-P. Xhonneux, M. Zhang, M. Gazeau, and J. Tang. “A*Net: A Scalable Path-based Reasoning Approach for Knowledge Graphs”. In: *NeurIPS* (2023).

[386] Z. Zhu, Z. Zhang, L.-P. Xhonneux, and J. Tang. “Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction”. In: *NeurIPS* (2021).

[387] F. Zollmann, J. Klaehn, T. Sikka, K. Comeforo, D. Broudy, M. Tröger, E. Poole, A. Edgley, and A. Mullen. “The Propaganda Model and Intersectionality: Integrating Separate Paradigms”. In: *Media Theory* (2018).

Summary

Narratives help humans make sense of experience, explain events, and establish shared understanding. Implementing such capabilities in AI can make systems more human-centric and aligned with how people reason, communicate, and explain. Beyond textual storytelling, narratives support a broad range of cognitive functions such as question answering, hypothesis generation, and discourse analysis, by linking new observations to background knowledge.

While large language models have demonstrated remarkable proficiency in generating fluent text, they face limitations such as hallucinations or verifiability. Knowledge-based approaches, particularly knowledge graphs, offer interpretability and structured reasoning. This thesis combines the strengths of both traditions to support narrative construction across heterogeneous domains, investigating how knowledge graphs enable narrative-like functions such as explanation, hypothesis generation, and discourse analysis.

Towards *representing* narrative-like structures as knowledge graphs, we first define narrative requirements and review ontological models for computational implementation. Practical guidelines are proposed for choosing suitable structured formats, which we apply to three use cases: historical explanation, analysis of social media discourse on inequality, generation of hypotheses in the field of social science. Despite differences in data formats and domain assumptions, these cases share a unified high-level approach.

We split the *construction* of narrative-centric knowledge graphs into retrieving relevant content and transforming it into a knowledge graph. We introduce a semantically guided retrieval method for identifying event-relevant subgraphs and apply domain-adapted strategies to build knowledge graphs across the three use cases.

We evaluate the *effectiveness* of structured narrative representations through quantitative and qualitative methods. In the historical domain, semantically guided retrieval improves content relevance, while overly complex narrative elements can hinder performance. Prompts enriched with event-centric knowledge graph improve factual accuracy in question answering without compromising succinctness. In the social science domain, AI-generated hypotheses, though not always surpassing human-generated ones, provide diverse options that complement and enhance the hypothesis generation process.

Our findings show that structured narrative representations can support both automated and human-centered reasoning tasks. The three use cases demonstrate broad applicability and adaptability across different domains. The methods presented here advance AI systems that engage more meaningfully with human reasoning, enabling narrative-aware tools to assist in answering questions and analyzing complex phenomena.

List of Figures

List of Tables

SIKS Dissertatiereeks

2016
01 Syed Saiden Abbas (RUN), Recognition of Shapes by Humans and Machines
02 Michiel Christiaan Meulendijk (UU), Optimizing medication reviews through decision support: prescribing a better pill to swallow
03 Maya Sappelli (RUN), Knowledge Work in Context: User Centered Knowledge Worker Support
04 Laurens Rietveld (VUA), Publishing and Consuming Linked Data
05 Evgeny Sherkhonov (UvA), Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers
06 Michel Wilson (TUD), Robust scheduling in an uncertain environment
07 Jeroen de Man (VUA), Measuring and modeling negative emotions for virtual training
08 Matje van de Camp (TiU), A Link to the Past: Constructing Historical Social Networks from Unstructured Data
09 Archana Nottamkandath (VUA), Trusting Crowdsourced Information on Cultural Artefacts
10 George Karafotias (VUA), Parameter Control for Evolutionary Algorithms
11 Anne Schuth (UvA), Search Engines that Learn from Their Users
12 Max Knobbout (UU), Logics for Modelling and Verifying Normative Multi-Agent Systems
13 Nana Baah Gyan (VUA), The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach

14 Ravi Khadka (UU), Revisiting Legacy Software System Modernization
15 Steffen Michels (RUN), Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments
16 Guangliang Li (UvA), Socially Intelligent Autonomous Agents that Learn from Human Reward
17 Berend Weel (VUA), Towards Embodied Evolution of Robot Organisms
18 Albert Meroño Peñuela (VUA), Refining Statistical Data on the Web
19 Julia Efremova (TU/e), Mining Social Structures from Genealogical Data
20 Daan Odijk (UvA), Context & Semantics in News & Web Search
21 Alejandro Moreno Célleri (UT), From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground
22 Grace Lewis (VUA), Software Architecture Strategies for Cyber-Foraging Systems
23 Fei Cai (UvA), Query Auto Completion in Information Retrieval
24 Brend Wanders (UT), Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach
25 Julia Kiseleva (TU/e), Using Contextual Information to Understand Searching and Browsing Behavior
26 Dilhan Thilakarathne (VUA), In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains
27 Wen Li (TUD), Understanding Geo-spatial Information on Social Media
28 Mingxin Zhang (TUD), Large-scale Agent-based Social Simulation - A study on epidemic prediction and control
29 Nicolas Höning (TUD), Peak reduction in decentralised electricity systems - Markets and prices for flexible planning
30 Ruud Mattheij (TiU), The Eyes Have It
31 Mohammad Khelghati (UT), Deep web content monitoring
32 Eelco Vriezekolk (UT), Assessing Telecommunication Service Availability Risks for Crisis Organisations
33 Peter Bloem (UvA), Single Sample Statistics, exercises in learning from just one example

	34	Dennis Schunselaar (TU/e), Configurable Process Trees: Elicitation, Analysis, and Enactment
	35	Zhaochun Ren (UvA), Monitoring Social Media: Summarization, Classification and Recommendation
	36	Daphne Karreman (UT), Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies
	37	Giovanni Sileno (UvA), Aligning Law and Action - a conceptual and computational inquiry
	38	Andrea Minuto (UT), Materials that Matter - Smart Materials meet Art & Interaction Design
	39	Merijn Bruijnes (UT), Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect
	40	Christian Detweiler (TUD), Accounting for Values in Design
	41	Thomas King (TUD), Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance
	42	Spyros Martzoukos (UvA), Combinatorial and Compositional Aspects of Bilingual Aligned Corpora
	43	Saskia Koldijk (RUN), Context-Aware Support for Stress Self-Management: From Theory to Practice
	44	Thibault Sellam (UvA), Automatic Assistants for Database Exploration
	45	Bram van de Laar (UT), Experiencing Brain-Computer Interface Control
	46	Jorge Gallego Perez (UT), Robots to Make you Happy
	47	Christina Weber (UL), Real-time foresight - Preparedness for dynamic innovation networks
	48	Tanja Buttler (TUD), Collecting Lessons Learned
	49	Gleb Polevoy (TUD), Participation and Interaction in Projects. A Game-Theoretic Analysis
	50	Yan Wang (TiU), The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains
2017	01	Jan-Jaap Oerlemans (UL), Investigating Cybercrime
	02	Sjoerd Timmer (UU), Designing and Understanding Forensic Bayesian Networks using Argumentation
	03	Daniël Harold Telgen (UU), Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines
	04	Mrunal Gawade (CWI), Multi-core Parallelism in a Column-store

05 Mahdieh Shadi (UvA), Collaboration Behavior
06 Damir Vandic (EUR), Intelligent Information Systems for Web Product Search
07 Roel Bertens (UU), Insight in Information: from Abstract to Anomaly
08 Rob Konijn (VUA), Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery
09 Dong Nguyen (UT), Text as Social and Cultural Data: A Computational Perspective on Variation in Text
10 Robby van Delden (UT), (Steering) Interactive Play Behavior
11 Florian Kunneman (RUN), Modelling patterns of time and emotion in Twitter #anticipointment
12 Sander Leemans (TU/e), Robust Process Mining with Guarantees
13 Gijs Huisman (UT), Social Touch Technology - Extending the reach of social touch through haptic technology
14 Shoshannah Tekofsky (TiU), You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior
15 Peter Berck (RUN), Memory-Based Text Correction
16 Aleksandr Chuklin (UvA), Understanding and Modeling Users of Modern Search Engines
17 Daniel Dimov (UL), Crowdsourced Online Dispute Resolution
18 Ridho Reinanda (UvA), Entity Associations for Search
19 Jeroen Vuurens (UT), Proximity of Terms, Texts and Semantic Vectors in Information Retrieval
20 Mohammadbashir Sedighi (TUD), Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility
21 Jeroen Linssen (UT), Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)
22 Sara Magliacane (VUA), Logics for causal inference under uncertainty
23 David Graus (UvA), Entities of Interest — Discovery in Digital Traces
24 Chang Wang (TUD), Use of Affordances for Efficient Robot Learning
25 Veruska Zamborlini (VUA), Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search

26 Merel Jung (UT), Socially intelligent robots that understand and respond to human touch
27 Michiel Joosse (UT), Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors
28 John Klein (VUA), Architecture Practices for Complex Contexts
29 Adel Alhuraibi (TiU), From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT"
30 Wilma Latuny (TiU), The Power of Facial Expressions
31 Ben Ruijl (UL), Advances in computational methods for QFT calculations
32 Thaer Samar (RUN), Access to and Retrievability of Content in Web Archives
33 Brigit van Loggem (OU), Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity
34 Maren Scheffel (OU), The Evaluation Framework for Learning Analytics
35 Martine de Vos (VUA), Interpreting natural science spreadsheets
36 Yuanhao Guo (UL), Shape Analysis for Phenotype Characterisation from High-throughput Imaging
37 Alejandro Montes Garcia (TU/e), WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy
38 Alex Kayal (TUD), Normative Social Applications
39 Sara Ahmadi (RUN), Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR
40 Altaf Hussain Abro (VUA), Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems
41 Adnan Manzoor (VUA), Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle
42 Elena Sokolova (RUN), Causal discovery from mixed and missing data with applications on ADHD datasets
43 Maaike de Boer (RUN), Semantic Mapping in Video Retrieval
44 Garm Lucassen (UU), Understanding User Stories - Computational Linguistics in Agile Requirements Engineering
45 Bas Testerink (UU), Decentralized Runtime Norm Enforcement
46 Jan Schneider (OU), Sensor-based Learning Support
47 Jie Yang (TUD), Crowd Knowledge Creation Acceleration

	48	Angel Suarez (OU), Collaborative inquiry-based learning
2018	01	Han van der Aa (VUA), Comparing and Aligning Process Representations
	02	Felix Mannhardt (TU/e), Multi-perspective Process Mining
	03	Steven Bosems (UT), Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction
	04	Jordan Janeiro (TUD), Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks
	05	Hugo Huurdeman (UvA), Supporting the Complex Dynamics of the Information Seeking Process
	06	Dan Ionita (UT), Model-Driven Information Security Risk Assessment of Socio-Technical Systems
	07	Jieting Luo (UU), A formal account of opportunism in multi-agent systems
	08	Rick Smetsers (RUN), Advances in Model Learning for Software Systems
	09	Xu Xie (TUD), Data Assimilation in Discrete Event Simulations
	10	Julienka Mollee (VUA), Moving forward: supporting physical activity behavior change through intelligent technology
	11	Mahdi Sargolzaei (UvA), Enabling Framework for Service-oriented Collaborative Networks
	12	Xixi Lu (TU/e), Using behavioral context in process mining
	13	Seyed Amin Tabatabaei (VUA), Computing a Sustainable Future
	14	Bart Joosten (TiU), Detecting Social Signals with Spatiotemporal Gabor Filters
	15	Naser Davarzani (UM), Biomarker discovery in heart failure
	16	Jaebok Kim (UT), Automatic recognition of engagement and emotion in a group of children
	17	Jianpeng Zhang (TU/e), On Graph Sample Clustering
	18	Henriette Nakad (UL), De Notaris en Private Rechtspraak
	19	Minh Duc Pham (VUA), Emergent relational schemas for RDF
	20	Manxia Liu (RUN), Time and Bayesian Networks
	21	Aad Slootmaker (OU), EMERGO: a generic platform for authoring and playing scenario-based serious games
	22	Eric Fernandes de Mello Araújo (VUA), Contagious: Modeling the Spread of Behaviours, Perceptions and Emotions in Social Networks
	23	Kim Schouten (EUR), Semantics-driven Aspect-Based Sentiment Analysis

	24	Jered Vroon (UT), Responsive Social Positioning Behaviour for Semi-Autonomous Telepresence Robots
	25	Riste Gligorov (VUA), Serious Games in Audio-Visual Collections
	26	Roelof Anne Jelle de Vries (UT),Theory-Based and Tailor-Made: Motivational Messages for Behavior Change Technology
	27	Maikel Leemans (TU/e), Hierarchical Process Mining for Scalable Software Analysis
	28	Christian Willemse (UT), Social Touch Technologies: How they feel and how they make you feel
	29	Yu Gu (TiU), Emotion Recognition from Mandarin Speech
	30	Wouter Beek (VUA), The "K" in "semantic web" stands for "knowledge": scaling semantics to the web
2019	01	Rob van Eijk (UL),Web privacy measurement in real-time bidding systems. A graph-based approach to RTB system classification
	02	Emmanuelle Beauxis Aussalet (CWI, UU), Statistics and Visualizations for Assessing Class Size Uncertainty
	03	Eduardo Gonzalez Lopez de Murillas (TU/e), Process Mining on Databases: Extracting Event Data from Real Life Data Sources
	04	Ridho Rahmadi (RUN), Finding stable causal structures from clinical data
	05	Sebastiaan van Zelst (TU/e), Process Mining with Streaming Data
	06	Chris Dijkshoorn (VUA), Nichesourcing for Improving Access to Linked Cultural Heritage Datasets
	07	Soude Fazeli (TUD), Recommender Systems in Social Learning Platforms
	08	Frits de Nijs (TUD), Resource-constrained Multi-agent Markov Decision Processes
	09	Fahimeh Alizadeh Moghaddam (UvA), Self-adaptation for energy efficiency in software systems
	10	Qing Chuan Ye (EUR), Multi-objective Optimization Methods for Allocation and Prediction
	11	Yue Zhao (TUD), Learning Analytics Technology to Understand Learner Behavioral Engagement in MOOCs
	12	Jacqueline Heinerman (VUA), Better Together
	13	Guanliang Chen (TUD), MOOC Analytics: Learner Modeling and Content Generation
	14	Daniel Davis (TUD), Large-Scale Learning Analytics: Modeling Learner Behavior & Improving Learning Outcomes in Massive Open Online Courses

15 Erwin Walraven (TUD), Planning under Uncertainty in Constrained and Partially Observable Environments
16 Guangming Li (TU/e), Process Mining based on Object-Centric Behavioral Constraint (OCBC) Models
17 Ali Hurriyetoglu (RUN),Extracting actionable information from microtexts
18 Gerard Wagenaar (UU), Artefacts in Agile Team Communication
19 Vincent Koeman (TUD), Tools for Developing Cognitive Agents
20 Chide Groenouwe (UU), Fostering technically augmented human collective intelligence
21 Cong Liu (TU/e), Software Data Analytics: Architectural Model Discovery and Design Pattern Detection
22 Martin van den Berg (VUA),Improving IT Decisions with Enterprise Architecture
23 Qin Liu (TUD), Intelligent Control Systems: Learning, Interpreting, Verification
24 Anca Dumitrache (VUA), Truth in Disagreement - Crowdsourcing Labeled Data for Natural Language Processing
25 Emiel van Miltenburg (VUA), Pragmatic factors in (automatic) image description
26 Prince Singh (UT), An Integration Platform for Synchromodal Transport
27 Alessandra Antonaci (OU), The Gamification Design Process applied to (Massive) Open Online Courses
28 Esther Kuindersma (UL), Cleared for take-off: Game-based learning to prepare airline pilots for critical situations
29 Daniel Formolo (VUA), Using virtual agents for simulation and training of social skills in safety-critical circumstances
30 Vahid Yazdanpanah (UT), Multiagent Industrial Symbiosis Systems
31 Milan Jelisavcic (VUA), Alive and Kicking: Baby Steps in Robotics
32 Chiara Sironi (UM), Monte-Carlo Tree Search for Artificial General Intelligence in Games
33 Anil Yaman (TU/e), Evolution of Biologically Inspired Learning in Artificial Neural Networks
34 Negar Ahmadi (TU/e), EEG Microstate and Functional Brain Network Features for Classification of Epilepsy and PNES
35 Lisa Facey-Shaw (OU), Gamification with digital badges in learning programming

	36	Kevin Ackermans (OU), Designing Video-Enhanced Rubrics to Master Complex Skills
	37	Jian Fang (TUD), Database Acceleration on FPGAs
	38	Akos Kadar (OU), Learning visually grounded and multilingual representations
2020	01	Armon Toubman (UL), Calculated Moves: Generating Air Combat Behaviour
	02	Marcos de Paula Bueno (UL), Unraveling Temporal Processes using Probabilistic Graphical Models
	03	Mostafa Deghani (UvA), Learning with Imperfect Supervision for Language Understanding
	04	Maarten van Gompel (RUN), Context as Linguistic Bridges
	05	Yulong Pei (TU/e), On local and global structure mining
	06	Preethu Rose Anish (UT), Stimulation Architectural Thinking during Requirements Elicitation - An Approach and Tool Support
	07	Wim van der Vegt (OU), Towards a software architecture for reusable game components
	08	Ali Mirsoleimani (UL),Structured Parallel Programming for Monte Carlo Tree Search
	09	Myriam Traub (UU), Measuring Tool Bias and Improving Data Quality for Digital Humanities Research
	10	Alifah Syamsiyah (TU/e), In-database Preprocessing for Process Mining
	11	Sepideh Mesbah (TUD), Semantic-Enhanced Training Data AugmentationMethods for Long-Tail Entity Recognition Models
	12	Ward van Breda (VUA), Predictive Modeling in E-Mental Health: Exploring Applicability in Personalised Depression Treatment
	13	Marco Virgolin (CWI), Design and Application of Gene-pool Optimal Mixing Evolutionary Algorithms for Genetic Programming
	14	Mark Raasveldt (CWI/UL), Integrating Analytics with Relational Databases
	15	Konstantinos Georgiadis (OU), Smart CAT: Machine Learning for Configurable Assessments in Serious Games
	16	Ilona Wilmont (RUN), Cognitive Aspects of Conceptual Modelling
	17	Daniele Di Mitri (OU), The Multimodal Tutor: Adaptive Feedback from Multimodal Experiences
	18	Georgios Methenitis (TUD), Agent Interactions & Mechanisms in Markets with Uncertainties: Electricity Markets in Renewable Energy Systems

	19	Guido van Capelleveen (UT), Industrial Symbiosis Recommender Systems
	20	Albert Hankel (VUA), Embedding Green ICT Maturity in Organisations
	21	Karine da Silva Miras de Araujo (VUA), Where is the robot?: Life as it could be
	22	Maryam Masoud Khamis (RUN), Understanding complex systems implementation through a modeling approach: the case of e-government in Zanzibar
	23	Rianne Conijn (UT), The Keys to Writing: A writing analytics approach to studying writing processes using keystroke logging
	24	Lenin da Nóbrega Medeiros (VUA/RUN), How are you feeling, human? Towards emotionally supportive chatbots
	25	Xin Du (TU/e), The Uncertainty in Exceptional Model Mining
	26	Krzysztof Leszek Sadowski (UU), GAMBIT: Genetic Algorithm for Model-Based mixed-Integer opTimization
	27	Ekaterina Muravyeva (TUD), Personal data and informed consent in an educational context
	28	Bibeg Limbu (TUD), Multimodal interaction for deliberate practice: Training complex skills with augmented reality
	29	Ioan Gabriel Bucur (RUN), Being Bayesian about Causal Inference
	30	Bob Zadok Blok (UL), Creatief, Creatiever, Creatiefst
	31	Gongjin Lan (VUA), Learning better – From Baby to Better
	32	Jason Rhuggenaath (TU/e), Revenue management in online markets: pricing and online advertising
	33	Rick Gilsing (TU/e), Supporting service-dominant business model evaluation in the context of business model innovation
	34	Anna Bon (UM), Intervention or Collaboration? Redesigning Information and Communication Technologies for Development
	35	Siamak Farshidi (UU), Multi-Criteria Decision-Making in Software Production
2021	01	Francisco Xavier Dos Santos Fonseca (TUD),Location-based Games for Social Interaction in Public Space
	02	Rijk Mercuur (TUD), Simulating Human Routines: Integrating Social Practice Theory in Agent-Based Models
	03	Seyyed Hadi Hashemi (UvA), Modeling Users Interacting with Smart Devices
	04	Ioana Jivet (OU), The Dashboard That Loved Me: Designing adaptive learning analytics for self-regulated learning

05 Davide Dell'Anna (UU), Data-Driven Supervision of Autonomous Systems
06 Daniel Davison (UT), "Hey robot, what do you think?" How children learn with a social robot
07 Armel Lefebvre (UU), Research data management for open science
08 Nardie Fanchamps (OU), The Influence of Sense-Reason-Act Programming on Computational Thinking
09 Cristina Zaga (UT), The Design of Robothings. Non-Anthropomorphic and Non-Verbal Robots to Promote Children's Collaboration Through Play
10 Quinten Meertens (UvA), Misclassification Bias in Statistical Learning
11 Anne van Rossum (UL), Nonparametric Bayesian Methods in Robotic Vision
12 Lei Pi (UL), External Knowledge Absorption in Chinese SMEs
13 Bob R. Schadenberg (UT), Robots for Autistic Children: Understanding and Facilitating Predictability for Engagement in Learning
14 Negin Samaeemofrad (UL), Business Incubators: The Impact of Their Support
15 Onat Ege Adali (TU/e), Transformation of Value Propositions into Resource Re-Configurations through the Business Services Paradigm
16 Esam A. H. Ghaleb (UM), Bimodal emotion recognition from audio-visual cues
17 Dario Dotti (UM), Human Behavior Understanding from motion and bodily cues using deep neural networks
18 Remi Wieten (UU), Bridging the Gap Between Informal Sense-Making Tools and Formal Systems - Facilitating the Construction of Bayesian Networks and Argumentation Frameworks
19 Roberto Verdecchia (VUA), Architectural Technical Debt: Identification and Management
20 Masoud Mansoury (TU/e), Understanding and Mitigating Multi-Sided Exposure Bias in Recommender Systems
21 Pedro Thiago Timbó Holanda (CWI), Progressive Indexes
22 Sihang Qiu (TUD), Conversational Crowdsourcing
23 Hugo Manuel Proença (UL), Robust rules for prediction and description
24 Kaijie Zhu (TU/e), On Efficient Temporal Subgraph Query Processing

	25	Eoin Martino Grua (VUA), The Future of E-Health is Mobile: Combining AI and Self-Adaptation to Create Adaptive E-Health Mobile Applications
	26	Benno Kruit (CWI/VUA), Reading the Grid: Extending Knowledge Bases from Human-readable Tables
	27	Jelte van Waterschoot (UT), Personalized and Personal Conversations: Designing Agents Who Want to Connect With You
	28	Christoph Selig (UL), Understanding the Heterogeneity of Corporate Entrepreneurship Programs
2022	01	Judith van Stegeren (UT), Flavor text generation for role-playing video games
	02	Paulo da Costa (TU/e), Data-driven Prognostics and Logistics Optimisation: A Deep Learning Journey
	03	Ali el Hassouni (VUA), A Model A Day Keeps The Doctor Away: Reinforcement Learning For Personalized Healthcare
	04	Ünal Aksu (UU), A Cross-Organizational Process Mining Framework
	05	Shiwei Liu (TU/e), Sparse Neural Network Training with In-Time Over-Parameterization
	06	Reza Refaei Afshar (TU/e), Machine Learning for Ad Publishers in Real Time Bidding
	07	Sambit Praharaj (OU), Measuring the Unmeasurable? Towards Automatic Co-located Collaboration Analytics
	08	Maikel L. van Eck (TU/e), Process Mining for Smart Product Design
	09	Oana Andreea Inel (VUA), Understanding Events: A Diversity-driven Human-Machine Approach
	10	Felipe Moraes Gomes (TUD), Examining the Effectiveness of Collaborative Search Engines
	11	Mirjam de Haas (UT), Staying engaged in child-robot interaction, a quantitative approach to studying preschoolers' engagement with robots and tasks during second-language tutoring
	12	Guanyi Chen (UU), Computational Generation of Chinese Noun Phrases
	13	Xander Wilcke (VUA), Machine Learning on Multimodal Knowledge Graphs: Opportunities, Challenges, and Methods for Learning on Real-World Heterogeneous and Spatially-Oriented Knowledge
	14	Michiel Overeem (UU), Evolution of Low-Code Platforms
	15	Jelmer Jan Koorn (UU), Work in Process: Unearthing Meaning using Process Mining

16 Pieter Gijsbers (TU/e), Systems for AutoML Research
17 Laura van der Lubbe (VUA), Empowering vulnerable people with serious games and gamification
18 Paris Mavromoustakos Blom (TiU), Player Affect Modelling and Video Game Personalisation
19 Bilge Yigit Ozkan (UU), Cybersecurity Maturity Assessment and Standardisation
20 Fakhra Jabeen (VUA), Dark Side of the Digital Media - Computational Analysis of Negative Human Behaviors on Social Media
21 Seethu Mariyam Christopher (UM), Intelligent Toys for Physical and Cognitive Assessments
22 Alexandra Sierra Rativa (TiU), Virtual Character Design and its potential to foster Empathy, Immersion, and Collaboration Skills in Video Games and Virtual Reality Simulations
23 Ilir Kola (TUD), Enabling Social Situation Awareness in Support Agents
24 Samaneh Heidari (UU), Agents with Social Norms and Values - A framework for agent based social simulations with social norms and personal values
25 Anna L.D. Latour (UL), Optimal decision-making under constraints and uncertainty
26 Anne Dirkson (UL), Knowledge Discovery from Patient Forums: Gaining novel medical insights from patient experiences
27 Christos Athanasiadis (UM), Emotion-aware cross-modal domain adaptation in video sequences
28 Onuralp Ulusoy (UU), Privacy in Collaborative Systems
29 Jan Kolkmeier (UT), From Head Transform to Mind Transplant: Social Interactions in Mixed Reality
30 Dean De Leo (CWI), Analysis of Dynamic Graphs on Sparse Arrays
31 Konstantinos Traganos (TU/e), Tackling Complexity in Smart Manufacturing with Advanced Manufacturing Process Management
32 Cezara Pastrav (UU), Social simulation for socio-ecological systems
33 Brinn Hekkelman (CWI/TUD), Fair Mechanisms for Smart Grid Congestion Management
34 Nimat Ullah (VUA), Mind Your Behaviour: Computational Modelling of Emotion & Desire Regulation for Behaviour Change

	35	Mike E.U. Ligthart (VUA), Shaping the Child-Robot Relationship: Interaction Design Patterns for a Sustainable Interaction
2023	01	Bojan Simoski (VUA), Untangling the Puzzle of Digital Health Interventions
	02	Mariana Rachel Dias da Silva (TiU), Grounded or in flight? What our bodies can tell us about the whereabouts of our thoughts
	03	Shabnam Najafian (TUD), User Modeling for Privacy-preserving Explanations in Group Recommendations
	04	Gineke Wiggers (UL), The Relevance of Impact: bibliometric-enhanced legal information retrieval
	05	Anton Bouter (CWI), Optimal Mixing Evolutionary Algorithms for Large-Scale Real-Valued Optimization, Including Real-World Medical Applications
	06	António Pereira Barata (UL), Reliable and Fair Machine Learning for Risk Assessment
	07	Tianjin Huang (TU/e), The Roles of Adversarial Examples on Trustworthiness of Deep Learning
	08	Lu Yin (TU/e), Knowledge Elicitation using Psychometric Learning
	09	Xu Wang (VUA), Scientific Dataset Recommendation with Semantic Techniques
	10	Dennis J.N.J. Soemers (UM), Learning State-Action Features for General Game Playing
	11	Fawad Taj (VUA), Towards Motivating Machines: Computational Modeling of the Mechanism of Actions for Effective Digital Health Behavior Change Applications
	12	Tessel Bogaard (VUA), Using Metadata to Understand Search Behavior in Digital Libraries
	13	Injy Sarhan (UU), Open Information Extraction for Knowledge Representation
	14	Selma Čaušević (TUD), Energy resilience through self-organization
	15	Alvaro Henrique Chaim Correia (TU/e), Insights on Learning Tractable Probabilistic Graphical Models
	16	Peter Blomsma (TiU), Building Embodied Conversational Agents: Observations on human nonverbal behaviour as a resource for the development of artificial characters
	17	Meike Nauta (UT), Explainable AI and Interpretable Computer Vision – From Oversight to Insight

	18	Gustavo Penha (TUD), Designing and Diagnosing Models for Conversational Search and Recommendation
	19	George Aalbers (TiU), Digital Traces of the Mind: Using Smartphones to Capture Signals of Well-Being in Individuals
	20	Arkadiy Dushatskiy (TUD), Expensive Optimization with Model-Based Evolutionary Algorithms applied to Medical Image Segmentation using Deep Learning
	21	Gerrit Jan de Bruin (UL), Network Analysis Methods for Smart Inspection in the Transport Domain
	22	Alireza Shojaifar (UU), Volitional Cybersecurity
	23	Theo Theunissen (UU), Documentation in Continuous Software Development
	24	Agathe Balayn (TUD), Practices Towards Hazardous Failure Diagnosis in Machine Learning
	25	Jurian Baas (UU), Entity Resolution on Historical Knowledge Graphs
	26	Loek Tonnaer (TU/e), Linearly Symmetry-Based Disentangled Representations and their Out-of-Distribution Behaviour
	27	Ghada Sokar (TU/e), Learning Continually Under Changing Data Distributions
	28	Floris den Hengst (VUA), Learning to Behave: Reinforcement Learning in Human Contexts
	29	Tim Draws (TUD), Understanding Viewpoint Biases in Web Search Results
2024	01	Daphne Miedema (TU/e), On Learning SQL: Disentangling concepts in data systems education
	02	Emile van Krieken (VUA), Optimisation in Neurosymbolic Learning Systems
	03	Feri Wijayanto (RUN), Automated Model Selection for Rasch and Mediation Analysis
	04	Mike Huisman (UL), Understanding Deep Meta-Learning
	05	Yiyong Gou (UM), Aerial Robotic Operations: Multi-environment Cooperative Inspection & Construction Crack Autonomous Repair
	06	Azqa Nadeem (TUD), Understanding Adversary Behavior via XAI: Leveraging Sequence Clustering to Extract Threat Intelligence
	07	Parisa Shayan (TiU), Modeling User Behavior in Learning Management Systems

08 Xin Zhou (UvA), From Empowering to Motivating: Enhancing Policy Enforcement through Process Design and Incentive Implementation
09 Giso Dal (UT), Probabilistic Inference Using Partitioned Bayesian Networks
10 Cristina-Iulia Bucur (VUA), Linkflows: Towards Genuine Semantic Publishing in Science
11 withdrawn
12 Peide Zhu (TUD), Towards Robust Automatic Question Generation For Learning
13 Enrico Liscio (TUD), Context-Specific Value Inference via Hybrid Intelligence
14 Larissa Capobianco Shimomura (TU/e), On Graph Generating Dependencies and their Applications in Data Profiling
15 Ting Liu (VUA), A Gut Feeling: Biomedical Knowledge Graphs for Interrelating the Gut Microbiome and Mental Health
16 Arthur Barbosa Câmara (TUD), Designing Search-as-Learning Systems
17 Razieh Alidoosti (VUA), Ethics-aware Software Architecture Design
18 Laurens Stoop (UU), Data Driven Understanding of Energy-Meteorological Variability and its Impact on Energy System Operations
19 Azadeh Mozafari Mehr (TU/e), Multi-perspective Conformance Checking: Identifying and Understanding Patterns of Anomalous Behavior
20 Ritsart Anne Plantenga (UL), Omgang met Regels
21 Federica Vinella (UU), Crowdsourcing User-Centered Teams
22 Zeynep Ozturk Yurt (TU/e), Beyond Routine: Extending BPM for Knowledge-Intensive Processes with Controllable Dynamic Contexts
23 Jie Luo (VUA), Lamarck's Revenge: Inheritance of Learned Traits Improves Robot Evolution
24 Nirmal Roy (TUD), Exploring the effects of interactive interfaces on user search behaviour
25 Alisa Rieger (TUD), Striving for Responsible Opinion Formation in Web Search on Debated Topics
26 Tim Gubner (CWI), Adaptively Generating Heterogeneous Execution Strategies using the VOILA Framework

27 Lincen Yang (UL), Information-theoretic Partition-based Models for Interpretable Machine Learning
28 Leon Helwerda (UL), Grip on Software: Understanding development progress of Scrum sprints and backlogs
29 David Wilson Romero Guzman (VUA), The Good, the Efficient and the Inductive Biases: Exploring Efficiency in Deep Learning Through the Use of Inductive Biases
30 Vijanti Ramautar (UU), Model-Driven Sustainability Accounting
31 Ziyu Li (TUD), On the Utility of Metadata to Optimize Machine Learning Workflows
32 Vinicius Stein Dani (UU), The Alpha and Omega of Process Mining
33 Siddharth Mehrotra (TUD), Designing for Appropriate Trust in Human-AI interaction
34 Robert Deckers (VUA), From Smallest Software Particle to System Specification - MuDForM: Multi-Domain Formalization Method
35 Sicui Zhang (TU/e), Methods of Detecting Clinical Deviations with Process Mining: a fuzzy set approach
36 Thomas Mulder (TU/e), Optimization of Recursive Queries on Graphs
37 James Graham Nevin (UvA), The Ramifications of Data Handling for Computational Models
38 Christos Koutras (TUD), Tabular Schema Matching for Modern Settings
39 Paola Lara Machado (TU/e), The Nexus between Business Models and Operating Models: From Conceptual Understanding to Actionable Guidance
40 Montijn van de Ven (TU/e), Guiding the Definition of Key Performance Indicators for Business Models
41 Georgios Siachamis (TUD), Adaptivity for Streaming Dataflow Engines
42 Emmeke Veltmeijer (VUA), Small Groups, Big Insights: Understanding the Crowd through Expressive Subgroup Analysis
43 Cedric Waterschoot (KNAW Meertens Instituut), The Constructive Conundrum: Computational Approaches to Facilitate Constructive Commenting on Online News Platforms
44 Marcel Schmitz (OU), Towards learning analytics-supported learning design
45 Sara Salimzadeh (TUD), Living in the Age of AI: Understanding Contextual Factors that Shape Human-AI Decision-Making

	46	Georgios Stathis (Leiden University), Preventing Disputes: Preventive Logic, Law & Technology
	47	Daniel Daza (VUA), Exploiting Subgraphs and Attributes for Representation Learning on Knowledge Graphs
	48	Ioannis Petros Samiotis (TUD), Crowd-Assisted Annotation of Classical Music Compositions
2025	01	Max van Haastrecht (UL), Transdisciplinary Perspectives on Validity: Bridging the Gap Between Design and Implementation for Technology-Enhanced Learning Systems
	02	Jurgen van den Hoogen (JADS), Time Series Analysis Using Convolutional Neural Networks
	03	Andra-Denis Ionescu (TUD), Feature Discovery for Data-Centric AI
	04	Rianne Schouten (TU/e), Exceptional Model Mining for Hierarchical Data
	05	Nele Albers (TUD), Psychology-Informed Reinforcement Learning for Situated Virtual Coaching in Smoking Cessation
	06	Daniël Vos (TUD), Decision Tree Learning: Algorithms for Robust Prediction and Policy Optimization
	07	Ricky Maulana Fajri (TU/e), Towards Safer Active Learning: Dealing with Unwanted Biases, Graph-Structured Data, Adversary, and Data Imbalance
	08	Stefan Bloemheuvel (TiU), Spatio-Temporal Analysis Through Graphs: Predictive Modeling and Graph Construction
	09	Fadime Kaya (VUA), Decentralized Governance Design - A Model-Based Approach
	10	Zhao Yang (UL), Enhancing Autonomy and Efficiency in Goal-Conditioned Reinforcement Learning
	11	Shahin Sharifi Noorian (TUD), From Recognition to Understanding: Enriching Visual Models Through Multi-Modal Semantic Integration
	12	Lijun Lyu (TUD), Interpretability in Neural Information Retrieval
	13	Fuda van Diggelen (VUA), Robots Need Some Education: on the complexity of learning in evolutionary robotics
	14	Gennaro Gala (TU/e), Probabilistic Generative Modeling with Latent Variable Hierarchies
	15	Michiel van der Meer (UL), Opinion Diversity through Hybrid Intelligence

16 Monika Grewal (TU Delft), Deep Learning for Landmark Detection, Segmentation, and Multi-Objective Deformable Registration in Medical Imaging
17 Matteo De Carlo (VUA), Real Robot Reproduction: Towards Evolving Robotic Ecosystems
18 Anouk Neerincx (UU), Robots That Care: How Social Robots Can Boost Children's Mental Wellbeing
19 Fang Hou (UU), Trust in Software Ecosystems
20 Alexander Melchior (UU), Modelling for Policy is More Than Policy Modelling (The Useful Application of Agent-Based Modelling in Complex Policy Processes)
21 Mandani Ntekouli (UM), Bridging Individual and Group Perspectives in Psychopathology: Computational Modeling Approaches using Ecological Momentary Assessment Data
22 Hilde Weerts (TU/e), Decoding Algorithmic Fairness: Towards Interdisciplinary Understanding of Fairness and Discrimination in Algorithmic Decision-Making
23 Roderick van der Weerdt (VUA), IoT Measurement Knowledge Graphs: Constructing, Working and Learning with IoT Measurement Data as a Knowledge Graph
24 Zhong Li (UL), Trustworthy Anomaly Detection for Smart Manufacturing
25 Kyana van Eijndhoven (TiU), A Breakdown of Breakdowns: Multi-Level Team Coordination Dynamics under Stressful Conditions
26 Tom Pepels (UM), Monte-Carlo Tree Search is Work in Progress
27 Danil Provodin (JADS, TU/e), Sequential Decision Making Under Complex Feedback
28 Jinke He (TU Delft), Exploring Learned Abstract Models for Efficient Planning and Learning
29 Erik van Haeringen (VUA), Mixed Feelings: Simulating Emotion Contagion in Groups
30 Myrthe Reuver (VUA), A Puzzle of Perspectives: Interdisciplinary Language Technology for Responsible News Recommendation
31 Gebrekirstos Gebreselassie Gebremeskel (RUN), Spotlight on Recommender Systems: Contributions to Selected Components in the Recommendation Pipeline
32 Ryan Brate (UU), Words Matter: A Computational Toolkit for Charged Terms

33 Merle Reimann (VUA), Speaking the Same Language: Spoken Capability Communication in Human-Agent and Human-Robot Interaction
34 Eduard C. Groen (UU), Crowd-Based Requirements Engineering
35 Urja Khurana (VUA), From Concept To Impact: Toward More Robust Language Model Deployment
36 Anna Maria Wegmann (UU), Say the Same but Differently: Computational Approaches to Stylistic Variation and Paraphrasing
37 Chris Kamphuis (RUN), Exploring Relations and Graphs for Information Retrieval
38 Valentina Maccatrozzo (VUA), Break the Bubble: Semantic Patterns for Serendipity
39 Dimitrios Alivanistos (VUA), Knowledge Graphs & Transformers for Hypothesis Generation: Accelerating Scientific Discovery in the Era of Artificial Intelligence
40 Stefan Grafberger (UvA), Declarative Machine Learning Pipeline Management via Logical Query Plans
41 Mozhgan Vazifehdoostirani (TU/e), Leveraging Process Flexibility to Improve Process Outcome - From Descriptive Analytics to Actionable Insights
42 Margherita Martorana (VUA), Semantic Interpretation of Dataless Tables: a metadata-driven approach for findable, accessible, interoperable and reusable restricted access data
43 Krist Shingjergji (OU), Sense the Classroom - Using AI to Detect and Respond to Learning-Centered Affective States in Online Education
44 Robbert Reijnen (TU/e), Dynamic Algorithm Configuration for Machine Scheduling Using Deep Reinforcement Learning
45 Anjana Mohandas Sheeladevi (VUA), Occupant-Centric Energy Management: Balancing Privacy, Well-being and Sustainability in Smart Buildings
46 Ya Song (TU/e), Graph Neural Networks for Modeling Temporal and Spatial Dimensions in Industrial Decision-making
47 Tom Kouwenhoven (UL), Collaborative Meaning-Making. The Emergence of Novel Languages in Humans, Machines, and Human-Machine Interactions
48 Evy van Weelden (TiU), Integrating Virtual Reality and Neurophysiology in Flight Training
49 Selene Báez Santamaría (VUA), Knowledge-centered conversational agents with a drive to learn

	50	Lea Krause (VUA), Contextualising Conversational AI
	51	Jiaxu Zhao (TU/e), Understanding and Mitigating Unwanted Biases in Generative Language Models
	52	Qiao Xiao (TU/e), Model, Data and Communication Sparsity for Efficient Training of Neural Networks
	53	Gaole He (TUD), Towards Effective Human-AI Collaboration: Promoting Appropriate Reliance on AI Systems
	54	Go Sugimoto (VUA), MISSING LINKS Investigating the Quality of Linked Data and its Tools in Cultural Heritage and Digital Humanities
	55	Sietze Kai Kuilman (TUD), AI that Glitters is Not Gold: Requirements for Meaningful Control of AI Systems
	56	Wijnand van Woerkom (UU), A Fortiori Case-Based Reasoning: Formal Studies with Applications in Artificial Intelligence and Law
	57	Syeda Amna Sohail (UT), Privacy-Utility Trade-Off in Healthcare Metadata Sharing and Beyond: A Normative and Empirical Evaluation at Inter and Intra Organizational Levels
	58	Junhan Wen (TUD), "From iMage to Market": Machine-Learning-Empowered Fruit Supply
	59	Mohsen Abbaspour Onari (TU/e), From Explanation to Trust: Modeling and Measuring Trust in Explainable Decision Support
	60	Marcel Jurriaan Robeer (UU), Beyond Trust: A Causal Approach to Explainable AI in Law Enforcement
	61	Shuai Wang (VUA), Links in Large Integrated Knowledge Graphs: Analysis, Refinement, and Domain Applications
	62	Khaleel Asyraaf Mat Sanusi (OU), Augmenting a learning model within immersive learning environments for psychomotor skills
	63	Rashid Zaman (TU/e), Online Conformance Checking on Degraded Data
	64	Jens d'Hondt (TU/e), Effective and Efficient Multivariate Similarity Search
	65	Aswin Balasubramaniam (UT), Disentangling Runner Drone Interaction Potentialities
2026	01	Pei-Yu Chen (TUD), Human-Agent Alignment Dialogues: Eliciting User Information at Runtime for Personalized Behavior Support
	02	Hezha Hassan Mohammedkhan (TiU), Estimating Body Measurements of Children from 2D Images: Towards the Automatic Detection of Malnutrition

03 Kyriakos Psarakis (TUD), Democratizing Scalable Cloud Applications: Transactional Stateful Functions on Streaming Dataflows
04 Boyu Xu (UU), Exploring Indirect Relations Between Topics in Neuroscience Literature Using Augmented Reality to Inform Experimental Design
05 Koen Minartz (TU/e), Stochastic Simulation with Geometric Deep Generative Models
06 Azim Afroozeh (CWI, VUA), FastLanes: A Next-Gen File Format
07 Inès Blin (VUA), Narrative Understanding with Knowledge Graphs