University of Pisa
Department of Computer Science
PhD Program in Computer Science, XXXIII Cycle

Enhancing the Computational Representation of Narrative and Its Extraction from Text

Doctoral Thesis

Author
Daniele Metilli

Supervisors
Dr. Carlo Meghini
Prof. Maria Simi

External Reviewers
Prof. Béatrice Markhoff
Dr. Xenophon Zabulis

Coordinator of the PhD Program
Prof. Antonio Brogi

Pisa, October 2021

To my grandmother,
Anna Buschini (1925–2020),
and all other victims of COVID-19

Acknowledgements

The completion of this thesis would not have been possible without the invaluable help and support of my supervisors, Carlo Meghini and Maria Simi, whom I thank from the bottom of my heart.

I am grateful to the members of the internal committee, Vincenzo Gervasi, Valentina Bartalesi, and Felice Dell'Orletta, for their crucial advice; to the external reviewers, Béatrice Markhoff and Xenophon Zabulis, for their thorough and insightful work; and to the coordinators, Antonio Brogi and Paolo Ferragina, for their tireless support and guidance.

Thanks to all my colleagues at ISTI-CNR and at the University of Pisa who made my work possible during these trying times; and to the students of the Semantic Web course, from whom I have learned so much.

Thanks to my family, and especially my mother Franca, my sisters Elisa and Lucia, my lovely nieces Raffaella and Arianna, my amazing cousin Pietro, my aunt Clara, and my uncle Giorgio. A special thank you to Angel — you are still my family, and I will never stop caring about you.

Thanks to my friends, and above all Said and Nice, who helped me every step of the way. I couldn't have done this without you! Chiara, thank you for being such a good and caring friend, colleague, and kindred spirit — never stop fighting! Gaia, I will never forget your kindness, a ray of light in tough times. Luca, thanks for being an awesome friend and travel companion. Marta, I'm so grateful for your friendship, and for the dumplings (Wakame is cute!). Monica, you helped me immensely, thank you. Vale, I miss you, and can't wait to meet Elliot. Beatrice, I'm so proud of your accomplishments. Let's keep in touch!

Thanks to Alb, Alessio, Angelica, Cesare, Chichi, Elisabetta, Filippo, Francesca, Gianpaolo, Giulia, Giulio, Lorenzo, Luca, Marta, Matteo, Nicoletta, Nicolò, Orlando, Paola, the Queer Seminar, my Rainbowlist, and everyone who has been close to me throughout the PhD. Special thanks to all those who don't conform, who think freely, who speak up. You make the world a better place.

One last thought for Near, who was too good for this Earth and left early. Your work lives on, and you will not be forgotten.

Abstract

Narratives are a fundamental part of human life. Every human being encounters countless stories during their life, and these stories contribute to form a common understanding of reality. This is reflected in the current digital landscape, and especially on the Web, where narratives are published and shared everyday.

However, the current digital representation of narratives is limited by the fact that each narrative is generally expressed as natural language text or other media, in an unstructured way that is neither standardized nor machine-readable. These limitations hinder the manageability of narratives by automated systems.

One way to solve this problem would be to create an ontology of narrative, i.e., a formal model of what a narrative is, then develop semi-automated methods to extract narratives from natural language text, and use the extracted data to populate the ontology. However, the feasibility of this approach remains an open question.

This thesis attempts to investigate this research question, starting from the state of the art in the fields of Computational Narratology, Semantic Web, and Natural Language Processing. Based on this analysis, we have identified a set of requirements, and we have developed a methodology for our research work.

Then, we have developed an informal conceptualization of narrative, and we have expressed it in a formal way using First-Order Logic. The result of this work is the Narrative Ontology (NOnt), a formal model of narrative that also includes a representation of its textual structure and textual semantics. To ensure interoperability, the ontology is based on the CIDOC CRM and FRBRoo standards, and it has been expressed using the OWL and SWRL languages of the Semantic Web.

Based on the ontology, we have developed NarraNext, a semi-automatic tool that is able to extract the main elements of narrative from natural language text. The tool allows the user to create a complete narrative based on a text, using the extracted knowledge to populate the ontology. NarraNext is based on recent

advancements in the Natural Language Processing field, including deep neural networks, and is integrated with the Wikidata knowledge base.

The validation of our work is being carried out in three different scenarios: (i) a case study on biographies of historical figures found in Wikipedia; (ii) the Mingei project, which applies NOnt to the representation and preservation of Heritage Crafts; (iii) the Hypermedia Dante Network project, where NOnt has been integrated with a citation ontology to represent the content of Dante's Comedy.

All three applications have served to validate the representational adequacy of NOnt and the satisfaction of the requirements we defined. The case study on biographies has also evaluated the effectiveness of the NarraNext tool.

List of Publications

Journal Papers

1. Bartalesi, V., Meghini, C., & Metilli, D. (2017). A conceptualisation of narratives and its expression in the CRM. In: International Journal of Metadata, Semantics and Ontologies, vol. 12 (1), pp. 35–46. Inderscience Publishers. (https://www.inderscienceonline.com/doi/abs/10.1504/IJMSO.2017.087692)

2. Bartalesi, V., Meghini, C., Metilli, D., Tavoni, M., & Andriani, P. (2018). A web application for exploring primary sources: The DanteSources case study. In: Digital Scholarship in the Humanities, Oxford University Press. (https://academic.oup.com/dsh/article-abstract/33/4/705/4850504)

3. Meghini, C., Bartalesi, V., Metilli, D., & Benedetti F. (2019). Introducing narratives in Europeana: A case study. International Journal of Applied Mathematics and Computer Science, vol. 29 (1). University of Zielona Góra. (https://www.amcs.uz.zgora.pl/?action=paper&paper=1471)

4. Metilli, D., Bartalesi, V., & Meghini, C. (2019). A Wikidata-based tool for building and visualising narratives. International Journal on Digital Libraries, vol. 20 (1), pp. 417–432. Springer. (https://link.springer.com/article/10.1007/s00799-019-00266-3)

5. Zabulis, X., Meghini, C., Partarakis, N., Kaplanidi, D., Doulgeraki, P., Karuzaki, E., Stefanidi, E., Evdemon, T., Metilli, D., Bartalesi, V., Fasoula, M., Tasiopoulou, E., & Beisswenger, C. (2019). What is needed to digitise knowledge on heritage crafts? Memoriamedia Review (4). Memória Imaterial. (http://memoriamedia.net/pdfarticles/EN_MEMORIAMEDIA_REVIEW_Heritage_Crafts.pdf)

6. Zabulis, X., Meghini, C., Partarakis, N., Beisswenger, C., Dubois, A., Fasoula, M., Nitti, V., Ntoa, S., Adami, I., Chatziantoniou, A., Bartalesi, V., Metilli, D., Stivaktakis, N., Patsiouras, N., Doulgeraki, P., Karuzaki, E.,

Stefanidi, E., Qammaz, A., Kaplanidi, D., Neumann-Janßen, I., Denter, U., Hauser, H., Petraki, A., Stivaktakis, I., Mantinaki, E., Rigaki, A., & Galanakis, G. (2020). Representation and preservation of heritage crafts. Sustainability, vol. 12 (4). Multidisciplinary Digital Publishing Institute. (https://www.mdpi.com/2071-1050/12/4/1461/htm)

7. Meghini, C., Bartalesi, V., & Metilli, D. (2021). Representing Narratives in Digital Libraries: The Narrative Ontology. Semantic Web Journal, vol. 12 (2). IOS Press. (http://www.semantic-web-journal.net/content/representing-narratives-digital-libraries-narrative-ontology-1)

8. Partarakis, N., Kaplanidi, D., Doulgeraki, P., Karuzaki, E., Petraki, A., Metilli, D., Bartalesi, V., Adami, I., Meghini, C., & Zabulis, X. (2021). Representation and Presentation of Culinary tradition as Cultural Heritage. Heritage, vol. 4 (2), pp. 612-640. Multidisciplinary Digital Publishing Institute. (https://www.mdpi.com/2571-9408/4/2/36/htm)

9. Bartalesi, V., Metilli, D., Pratelli, N., & Pontari, P. (2021). Towards a Knowledge Base of Geographical Latin Works of Medieval and European Humanism: The IMAGO Ontology. Digital Scholarship in the Humanities. Oxford University Press. (https://doi.org/10.1093/llc/fqab060)

10. Partarakis, N., Doulgeraki, P., Karuzaki, E., Adami, I., Ntoa, S., Metilli, D., Bartalesi, V., , Meghini, C., Marketakis, Y., Theodoridou, M., Zabulis, X., & Kaplanidi, D. (forthcoming). Representation of socio-historical context to support the authoring and presentation of multimodal narratives: The Mingei Online Platform. Journal on Computing and Cultural Heritage. Association for Computing Machinery. Accepted for publication.

11. Bartalesi, V., Pratelli, N., Metilli, D., Tomazzoli, G., Livraghi, L., Zaccarello, M., & Meghini, C. (forthcoming). A Formal Representation of the Divine Comedy's Primary Sources: The Hypermedia Dante Network Ontology. Digital Scholarship in the Humanities. Oxford University Press. Accepted for publication.

Conference Papers

1. Tavoni, M., Andriani, P., Meghini, C., Bartalesi, V., & Metilli, D. (2017). L'esplorazione delle fonti dantesche attraverso la biblioteca digitale DanteSources. In: Thomas Persico & Riccardo Viel (eds.). Sulle tracce del Dante minore. Primo Convegno Bergomense di Studi su Dante Alighieri (Bergamo, Italy, 20–21 May 2016). Proceedings, pp. 29–52. Sestante Edizioni. (https://intranet.cnr.it/servizi/people/prodotto/scheda/i/404382)

2. Meghini, C., Bartalesi, V., Metilli, D., & Benedetti, F. (2018). A software architecture for narratives. In: Giuseppe Serra & Carlo Tasso (eds.). Digital Libraries and Multimedia Archives. IRCDL 2018 – 14th Italian Research Conference on Digital Libraries (Udine, Italy, 25–26 January 2018). Proceedings, pp. 23–29. Communications in Computer and Information

Science, vol. 806. Springer. (https://link.springer.com/chapter/10.1007/978-3-319-73165-0_3)

3. Bartalesi, V., Metilli, D., & Meghini, C. (2019). Constructing narratives using NBVT: A case study. In: 8th AIUCD Conference 2019 (Udine, Italy, 22–25 January 2019). Book of Abstracts. Associazione Italiana per l'Informatica Umanistica e la Cultura Digitale. (http://aiucd2019.uniud.it/wp-content/uploads/2020/03/AIUCD2019-BoA_DEF.pdf)

4. Metilli, D., Bartalesi, V., Meghini, C., & Aloia, N. (2019). Populating narratives using Wikidata events: An initial experiment. In: Digital Libraries: Supporting Open Science. ICRDL 2019 – 15th Italian Research Conference on Digital Libraries (Pisa, Italy, 31 January – 1 February 2019). Communications in Computer and Information Science, vol. 988. Springer. (https://link.springer.com/chapter/10.1007/978-3-030-11226-4_13)

5. Tomazzoli, G., Livraghi, L., Metilli, D., Pratelli, N., & Bartalesi, V. (2021). The Hypermedia Dante Network Project. In: 10th AIUCD Conference 2021 (Pisa, Italy, 19–21 January 2021). Book of Abstracts. Associazione Italiana per l'Informatica Umanistica e la Cultura Digitale. (https://aiucd2021.labcd.unipi.it/en/book-of-abstracts-conference)

Workshop Papers

1. Meghini, C., Bartalesi, V., Benedetti, F., & Metilli, D. (2019). Sistemi basati su ontologie per le Digital Humanities. In: Ital-IA 2019 – Workshop AI for Cultural Heritage (Rome, Italy, 18 March 2019). CINI National Lab. (http://www.ital-ia.it/submission/87/paper)

2. Metilli, D., Bartalesi, V., & Meghini, C. (2019). Steps towards a system to extract formal narratives from text. In: Text2Story 2019 – Second Workshop on Narrative Extraction From Texts (Cologne, Germany, 14 April 2019). Co-located with ECIR 2019. CEUR Workshop Proceedings, vol. 2342. CEUR-WS.org. (http://ceur-ws.org/Vol-2342/paper7.pdf)

Software

1. Metilli, D., Bartalesi, V., & Spadi, A. (2020). VDL Corpus Corporum API. ISTI-CNR. (http://vdl.isti.cnr.it/corpuscorporum.html)

2. Metilli, D., & Bartalesi, V. (2020). VDL DanteSearch API. ISTI-CNR. (http://vdl.isti.cnr.it/dantesearch.html)

3. Pratelli, N., Metilli, D., & Bartalesi, V. (2021). Hypermedia Dante Network Annotation Tool. ISTI-CNR. (https://hdn.dantenetwork.it/toolEN.html)

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

NARRATIVES are a fundamental part of human life. Every human being encounters countless stories in their everyday life: from the ones told by people in casual conversation, to the epic poems of the ancient past; from the suspenseful grip of a crime novel, to the tearful tale of a dramatic movie; from the fantastic stories invented by children playing in a park, to the historical memories recounted to them by their grandparents at dinner.

The reasons for this centrality of narrative in the human experience are complex and still under study. A widely-held thesis in psychology is that humans make sense of reality by structuring events into narrative (Bruner, 1991). As aptly put by Taylor (1989), "[a] basic condition of making sense of ourselves [is] that we grasp our lives in a narrative". Furthermore, it has long been known that narrative language is essential for successful communication between humans (Dautenhahn, 2002).

While humans are able to understand, create, and share narratives very easily, it is not the same when it comes to machines. At present, there is no way for a machine to interpret a narrative like a human does, and no machine is able to fully grasp what a narrative is, how it is structured, and what it means.

The present thesis reports a set of advancements in the field of computational modeling of narrative, including an ontology and a tool for narrative extraction from natural language text, that attempt to advance the current state of research in this field.

1.1 Background

Since the 1970s, much research has been carried out to study the computational representation of narratives (Mani, 2014). Despite significant efforts by researchers, to date there is still no standard way to digitally represent narratives, compare them, visualize them, or share them on the Web for further reuse.

One issue that has hindered progress in this field is the difficulty in defining exactly what a narrative is, because unfortunately a narrative is is a complex, multi-faceted object, and scholars have struggled to define it in a comprehensive way. According to Herman (2009), a narrative can be viewed "as a cognitive structure or way of making sense of experience, as a type of text, and as a resource for communicative interaction".

Other scholars view narrative in a more narrow way, e.g. "the representation of an event or of a sequence of events" (Genette & Lewin, 1983), or "a network of temporally-indexed representations of events" (Meister, 2003), focusing on the structural aspects of narrative, instead of its cognitive interpretation.

In the last few years, the Digital Libraries group at ISTI-CNR has started investigating narratives as important aspects of the contents of a Digital Library (DL) (Bartalesi, Meghini, & Metilli, 2017; Meghini et al., 2019). Every DL contains a set of digital objects, each endowed with some metadata that describe it. DLs also contain narratives in an embedded form, in the sense that their digital objects, their metadata, and their textual descriptions contain some elements of narrative.

For example, if the DL contains a depiction of Dante Alighieri writing the Divine Comedy, this digital object is referencing an event in the narrative of the poet's life. Similarly, other objects contained in the DL that are related to Dante will also be connected to this narrative, and by gathering these connections and representing them in a formal way it will be possible to reconstruct a (possibly partial) narrative about the life of the poet.

Narratives may thus be seen as a kind of "semantic glue" holding together the disparate digital objects that are scattered in the DL, often unconnected to one another. Furthermore, narratives may be useful even outside of traditional DLs, wherever there is a need to connect digital objects in a meaningful way to inform, educate, or entertain a user more effectively.

This need for a story-based layer that connects several resources and helps makes sense of a topic is by no means limited to DLs, In fact, it is at the basis of the recent interest in storytelling in the wider field of Digital Humanities (S. Wang & Zhan, 2010).

However, reaching this kind of wide application of narratives on the Web requires significant advancements in the digital representation of narrative. Working towards this goal, and following a knowledge-based approach, we began the development of a Narrative Ontology[1], i.e. a formal model to represent narratives (Bartalesi, Meghini, & Metilli, 2017) based on the CIDOC CRM standard ontology (Doerr, 2003).

[1]https://dlnarratives.eu/ontology.html

In the general definition that we have adopted as a basis for our ontology, a narrative is composed of a *fabula*, i.e. a network of events in chronological order, and a *plot*, i.e. a network of events selected from the fabula and partially ordered by the narrator (Herman, 2009; Propp, 2010). Each event is endowed with factual aspects (time, space, involved people or objects, etc.).

A narrative also includes a non-formal expression of the story, which can be a natural language text or other medium, e.g. audio or video (Bal, 1997). This aspect of narrative is called the *narration*, and it expresses the plot of the narrative.

On top of the ontology, we built a semi-automated Narrative Building and Visualising Tool (NBVT)[2], allowing users to construct and visualize narratives through a Web interface (Metilli, 2016). The tool is able to import knowledge from the Wikidata[3] knowledge base (Vrandei & Krötzsch, 2014), thereby facilitating the user in the construction of the narrative.

The existing ontology and tool can be used to represent any kind of narrative, be it real or fictional, except those that are inconsistent with physical reality (e.g. those in which effects precede causes, time forks, objects bilocate). However, both the ontology and tool currently have some significant limitations that hinder their wide application and deployment.

1.2 Research Proposal

The current version of the Narrative Ontology is able to successfully represent the fabula of the narrative, but the plot and the narration are modeled in a more limited way. In particular, the representation of the structural features of the text, of its semantics, and of the plot of the narrative, is very basic and can be improved. These limitations of the ontology make it very difficult to further automate the narrative-building process.

If the ontology were extended with a better representation of the text and plot, and if a tool was developed to successfully apply narrative extraction techniques to natural language text, new scenarios of application would open up. For instance, a user could load a textual biography into the tool and receive as output a set of candidate elements, such as events and related entities, that can be used to create a narrative about the subject of the biography.

The narrative building process would become significantly faster, allowing users to delegate to the machine the most tedious parts of the process and focus on the more creative side of narrative building. Many narratives could be created in a short amount of time, thereby opening up new and fascinating avenues of research involving large sets of formal narratives.

In order to achieve this vision, several steps need to be completed: (i) the ontology needs to be improved by devising better ways to represent the textual and plot levels of the narrative and their relationship with the fabula; (ii) a tool needs to be developed to extract narratives from text in a more automated way; (iii) both the ontology and the tool need to be validated through applications in

[2]https://dlnarratives.eu/tool.html
[3]https://wikidata.org

practical settings. These will be the three main lines of research of the present thesis.

Our key research questions can thus be expressed as follows:

Q1. How can we model the textual and plot levels of narrative, and their relationship with the fabula, in a better way?

Q2. How can we extract elements of narrative from text?

Q3. How can we validate this work, and which innovative applications can we develop based on it?

In the present thesis we have done our best to answer these questions, by working on:

A1. A new version of the Narrative Ontology that is better able to represent the textual and plot levels of narrative, including both structural and semantic aspects of natural language text.

A2. A narrative extraction system, called NarraNext, that is able to detect narrative elements in natural language text and allows a human user to build a narrative based on them.

A3. An evaluation on a set of biographies that validates our narrative extraction system, and in addition two further applications of our ontology in the context of Cultural Heritage and Digital Libraries.

Our work is at the intersection of several fields of Computer Science, including Artificial Intelligence (Russell et al., 2003), Semantic Web (Berners-Lee et al., 2001), and Natural Language Processing (C. D. Manning & Schütze, 1999). We incorporate methods and techniques from these fields into our research.

1.3 Outline of the Thesis

The thesis is structured as follows:

- Chapter 2 describes the state of the art in the fields of Narratology and Computational Narratology, focusing in particular on computational modeling of narrative and on narrative extraction from natural language text.
- Chapter 3 describes the methodology that we have followed, and presents a set of requirements that we have identified.
- Chapter 4 presents our conceptualization of narrative in an informal way.
- Chapter 5 describes the Narrative Ontology that we have developed based on the conceptualization.
- Chapter 6 presents NarraNext, a tool for creating narratives from natural lanuage text starting from a text in natural language.

- Chapter 7 reports an experimental validation of our narrative extraction system.
- Chapter 8 presents two additional applications of the Narrative Ontology.
- Finally, Chapter 9 reports our conclusions and future works.

CHAPTER 2

State of the Art

In this chapter we present the state of the art in the areas of Narratology and Computational Narratology. First of all, we will briefly introduce the concept of narrative and report how its definition has changed throughout the last century. Then, we will discuss the computational approaches to narrative representation from the 1970s to date, focusing in particular on ontologies to represent narrative and methods for narrative extraction. Finally, we will review some specific Computer Science techniques that can be applied to the task of narrative extraction, in particular from the field of Natural Language Processing.

2.1 Narratology

The discipline that studies the logic, principles, and practices of narrative representation is called *narratology* (Meister, 2014). Antecedents to it can be found as early as Aristotle's theory of aesthetics, where a narrative is defined as the imitation of real actions (*praxis*) that forms an argument (*logos*) whose fundamental units, or events, can be arranged in a plot (*mythos*) (Aristotle, 2013).

Modern narratology has its origin in the Russian formalism of the early 20th century. According to this school of thought, a universal pattern of codes, or common literary language, can be found in any narrative work, independently of the medium. The fundamental work of Propp (2010) proposed a model to represent folktales as combinations of basic building blocks, including thirty-one narrative functions and seven roles, or spheres of action, of the characters.

Russian formalism identifies two main elements of narrative: the *fabula*, defined as a series of events taking place at a certain time at a specific location, and the *syuzhet*, which is the particular way the story is narrated. While the order of the *fabula* is chronological, the order of the *syuzhet* corresponds to the way the events are presented by the author of the narrative (Propp, 2010; Shklovsky, 1997).

The other movement that greatly influenced the field of narratology in the 20th century is Structuralism. Drawing on Russian Formalism and on the work of De Saussure (1989), Levi-Strauss (1963) outlined a grammar of mythology. Todorov (1969) coined the term *narratologie*, and argued that narratives usually follow a path from an initial state of equilibrium to a different state of equilibrium, through a disruption phase.

Greimas et al. (1983) proposed a system of six basic structural elements of narratives called *actants*; Genette and Lewin (1983) codified a system of analysis that studied both the narration and the act of narrating, considering them separately from the story and content of the text. They also identified five concepts that characterize the syntax of narratives: order, frequency, duration, voice, and mood. Labov and Waletzky (1997) developed a model for the analysis of narratives in face-to-face interactions.

Unfortunately, no universally accepted definition of narrative exists yet (Ryan, 2007). For instance, in addition to the *fabula* and the *syuzhet*, Bal (1997) defines a third level that constitutes the concrete representation of the content that is conveyed to the audience (e.g., the text in a novel). On the other hand, Crawford (2012) posits that a narrative is a high-level structure based on causality, not on temporal or spatial relations.

Since the 1980s, post-structuralist perspectives on narratology have emerged. In particular, cognitive narratology considers the narrative as a psychological phenomenon, which should be studied from a cognitive perspective (Herman, 2000). A precursor to this approach can be found in Barthes (1966), who argued that the textual nature of a narrative should be studied together with its interpretation by the reader.

Herman (2009) proposes a definition of narrative based on four elements: (i) situatedness (the narrative is situated in a context), (ii) event sequencing, (iii) world disruption (the events introduce a disequilibrium in the world), and (iv) the experience of living through the story. Hyvärinen (2006) describes the spread of narrative across several disciplines, starting from the 1960s to the present day. This has been called by Kreiswirth (2000) the “narrative turn”.

2.2 Computational Narratology

Research on computational narratology started in the 1970s, when Artificial Intelligence (AI) researchers realized that narrative could be used to investigate and reproduce complex human understanding.

Computational narratology is the study of “the algorithmic processes involved in creating and interpreting narratives, modeling narrative structure in terms of formal computable representations” (Mani, 2012). The principal aim of computational narratology is to develop Artificial Intelligence systems for reproducing human-like narrative behaviour (Mani, 2014).

The group of Roger Schank at Yale University was the first to study narrative modeling through computer-based means (Schank & Abelson, 1975). Their research had two aims: (i) story understanding, i.e. the ability to process a natural language text and extract a narrative from it, and (ii) story generation, i.e. the ability to construct a human-like narrative. In the same period, Rumelhart (1975) developed the first general story grammar, based on the relations between sentences of a narrative text that make up the structure of the story.

The first story understanding systems were SAM (Cullingford, 1981) and PAM (Wilensky, 1981). These systems were rule-based, meaning that they used a set of rules defined by a human to interpret a text and extract a narrative from it. Similar systems from this era were CYRUS (Kolodner, 1980) and BORIS (Dyer, 1982). Story generation systems, such as (Meehan, 1976), were also rule-based. They started from basic building blocks and aimed to generate interesting narratives similar to those constructed by humans.

These first AI systems were fairly limited and required significant human input, but the approaches, problems encountered, and reasoning behind them are nevertheless of interest to us. Alas, this initial research on narrative understanding was not successful. It was soon clear to these researchers that the computational capabilities of the technology of the time were too limited to allow for general narrative understanding and generation.

After this initial bout of interest towards narrative in computer science, in the following two decades only a few researchers, e.g. Schank (1990), Ryan (1991), and Turner (1993), kept their focus on narrative. Despite some advances, the story understanding systems developed in this decade, such as AQUA (Ram, 1989) and ISAAC (Moorman & Ram, 1993), were not able overcome the limitations of the previous ones.

Then, at the end of the 1990s, there was a renewed interest in AI approaches to narrative. This was not a sudden event, but rather the consequence of technological developments in computing hardware, the emergence of the Web and the incorporation of narrative in other fields of computer science such as Human-Computer Interaction (Don, 1990; Laurel, 1991) and Social Network Analysis (Lawrence & Thomas, 1999; Sack, 2002).

In 1999, the Narrative Intelligence Symposium (Mateas & Sengers, 2003) gathered researchers from several different fields such as AI, psychology, and literary studies, setting the course for the subsequent decade of developments in the field. However, most of these works were centered on story generation (Bailey, 1999), interactive storytelling (Aylett, 1999), and narrative agents (Bickmore & Cassell, 1999), not on story understanding.

One of the most significant developments in this era was the application of cognitive science into AI studies on narrative, rooted in Constructivism (Wadsworth, 1996) and inspired by recent research in the human sciences, e.g. (Murray, 2017; Turkle, 1991), and in AI itself (Sengers, 1998). Several researchers analyzed the mechanisms of human understanding of narrative and attempted to reproduce them through AI, for example by modeling the psychological process of the narrative reader (Bailey, 1999; Cheong & Young, 2006), or the authorial process of the narrator (Szilas, 2007).

Around the turn of the century, the area of interactive digital storytelling (Szilas, 1999) became successful and started to diverge from the main research on computational narrative modeling, leading to the establishment of the ICIVS and TIDSE series of conferences, later unified into the Joint International Conference on Interactive Digital Storytelling (ICIDS) (Spierling & Szilas, 2008).

At the current stage of our research, works from this area of study are not particularly relevant to us, as they focus on interaction between a human user and a narrative environment. However, they may become relevant if, in the future, we set to develop applications featuring story generation or virtual environments.

At the same time, research on computational modeling of narrative moved forward. Comprehensive models of narrative were developed (Damiano et al., 2005; Lakoff & Narayanan, 2010), including models of story schemas, i.e. patterns of events that are typical of specific kinds of stories (Bex, 2011; Wagner, 2010). Semantic models of narrative informed continued research on story generation (Rishes et al., 2013), and newer applications such as story analogy and comparison (Elson, 2012).

The renewed interest in narrative led to the establishment of two series of workshops: Computational Models of Narrative (Finlayson et al., 2010) and Intelligent Narrative Technologies (Magerko & Riedl, 2007), which gathered the main researchers in the field and produced several interesting studies.

However, even in these fora, story understanding and narrative extraction remained a less popular topic of research for several years. One of the possible reasons for this absence is the belief that story understanding, in the sense intended by early AI researchers and later by cognitive narratologists, is impossible to achieve with current technology.

This view has been expressed by Mani (2014): "Story understanding systems (e.g. Wilensky 1978) never got very far, since (i) inferring characters goals involves a large search space and the inferences may need to be revised during processing and (ii) humans use a great deal of knowledge to interpret even simple stories".

We agree with the first assumption, but not with the second one: we do not believe that it is strictly necessary to represent the character's goals in a model of narrative. We prefer to focus on the events that happen in the narrative and on their temporal and causal succession, leaving the interpretation of the character's goals to the human user.

2.3 Semantic Web and Ontologies

In this section, we provide some background on the general concept of ontology in the Semantic Web field, and we provide a list of ontologies that are relevant to the present work.

First of all, the Semantic Web is a framework that aims to foster the sharing and reuse of knowledge by humans and machines alike (Berners-Lee et al., 2001). Through the technologies of the Semantic Web, it is possible to represent knowledge in a formal, machine-readable and standardized form. This knowledge can then be published on the Web following the Linked Data paradigm (Bizer et al., 2011).

The fundamental language of the Semantic Web is RDF (Resource Description Framework) (Manola & Miller, 2004), which allows the description of resources in the form of triples formed by a subject, a predicate, and an object.

RDF Schema (RDFS) (Brickley & Guha, 2014), which was developed on top of RDF, allow the definition of classes of resources, and also of hierarchies among classes and among properties, thereby allowing the construction of basic ontologies.

In this context, the term *ontology* describes the "formal, explicit specification of a shared conceptualization" (Guarino et al., 2009). This definition, suggested by Studer et al. (1998) and based on two previous definitions by Gruber (1993) and Borst (1999), highlights some important characteristics that an ontology must possess:

- an ontology is not simply a model of a certain domain of reality, but rather a *formal* model; in this it differs from a conceptualization, which is a more informal representation of knowledge; generally, the ontology *specifies* a conceptualization

- the underlying conceptualization must be *shared*, meaning that the ontology; cannot exist simply in the mind of a single person, but must be the result of a consensus between a group of people; indeed, ontologies are often used to standardize the; representation of a certain domain of reality

- an ontology must be *explicit*, meaning that it must be expressed and published through a formal language, such as logic or the languages of the Semantic Web.

OWL (Web Ontology Language) (W3C OWL Working Group, 2012) supports the definition of an ontology consisting of several kinds of terms, including individuals (basic resources), classes (sets of resources), properties expressing relations between resources, and axioms asserting facts that are known to be true in the model, for example restrictions or property characteristics.

The latest version of this language, OWL 2, has two main profiles: OWL 2 Full offers maximum expressiveness, but it is undecidable, while *OWL 2 DL* guarantees decidability by imposing some restrictions on expressiveness. Three subprofiles (OWL 2 EL, OWL 2 QL, OWL 2 RL) also exist that further sacrifice expressiveness to reduce the computational complexity of reasoning on the ontology.

2.3.1 Ontologies for Representing Narratives

Existing ontologies to represent narratives choose widely different approaches, based on the definition of narrative that is adopted by their authors. In the following, we report a set of recent computational models of narrative that are relevant to our study.

First of all, since some of these models are event-based, we cite some ontologies for the representation of events, such as the Event Ontology (Raimond & Abdallah, 2007), LODE (Linking Open Descriptions of Events) (Shaw et al., 2009), the Event-Model-F ontology (Scherp et al., 2009), and the Simple Event Model (SEM) (Van Hage et al., 2011). More general models that have been adopted in this field are the CIDOC CRM (Doerr et al., 2007), the Europeana Data Model (Doerr et al., 2010), and DOLCE (Gangemi et al., 2002).

Franzosi (1998) developed a set of linguistic and statistical tools, based on semantic grammars, that can be used by historians to study historical events (e.g. strikes, demonstrations and other types of collective conflict). The tools can be used to perform complex data analysis tasks, but the underlying model of narrative is quite limited and not tied to a standard ontology.

Elson and McKeown (2007) developed a general model of narrative that models semantics such as timelines, states, events, characters and goals. An annotation tool based on the model allows users to define formal propositions to represent spans of text, as well as temporal relations and other aspects of narratives, to generate story graphs. However, constructing a narrative requires significant user input, and the model is not based on a standard ontology.

Zarri (2009) devised a novel approach to the representation and management of narrative information, through the use of a newly defined language called Narrative Knowledge Representation Language (NKRL). This language is able represent the narratives themselves and also the templates, or story schemes, that constitute the overarching structures of narratives. The main limitation of this approach is the definition of a new non-standard language to represent narratives, incompatible with Semantic Web technologies.

Bex (2011) built a formal grammar of narrative to be applied in the context of criminal cases. This model of narrative is combined with a model of argumentation to make sense of the different narratives provided by witnesses of a crime. This is an interesting approach, but unfortunately it lacks expressiveness, because it reduces narratives to basic logical arguments.

Lombardo and Pizzo (2014) describe the Drammar ontology, a computational model of narrative based on the characters' intentions an annotation schema for the narrative features of media objects which relies on a formal theory of story and characters, along with the Cinematic tool for the annotation of video objects and the automatic editing of the annotated objects. However, understanding the intentions of the characters requires significant human intervention, and their interpretation may vary based on the annotator.

Storyspace (Wolff et al., 2012) is a software which allows the description of event-based stories from objects preserved in museums. It can be used to create multiple narratives from a single museum exhibition. The underlying model is specific to museums and not readily extensible to a more general representation of narrative.

Niehaus et al. (2014) introduce SONNET, a flexible narrative framework to integrate, select, and reuse narrative models, thereby lowering development costs and improving benefits from each model. The framework includes a lightweight ontology language for the definition of key terms and relationships among them, and it specifies model metadata in order to allow developers to discover and understand models more readily.

Eger et al. (2015) present Impulse, a novel representation of narratives built on top of predicate logic, including temporal representation of a story's actions and events and representation of the mental models of the story's characters. The drawback of this approach is that Impulse is computationally undecidable.

Tuominen, Hyvönen, et al. (2018) describe Bio CRM, a proposed extension of the CIDOC CRM that supports the construction of event-based biographical narratives. This model has been applied in the context of the Early Modern Letters Online (EMLO) project (Tuominen, Mäkelä, et al., 2018).

Finally, our Ontology for Narratives is a formal model to represent narratives (Bartalesi, Meghini, & Metilli, 2017), expressed in the OWL language of the Semantic Web (McGuinness & Van Harmelen, 2004). The ontology allows the semantic, formal representation of a narrative, intended as a sequence of events defined by a narrator. A semi-automated Narrative Building and Visualising Tool (NBVT) was built on top of the ontology, allowing users to construct and visualize narratives through a Web interface (Metilli, 2016). The tool has been applied in the Digital Libraries field in order to improve search and discovery functionalities (Meghini et al., 2017, 2019).

In our formal representation, a narrative consists of three main elements: (i) *fabula*, i.e. the sequence of events in chronological order, (ii) one or more *narrations*[1] that express the narrative, and (iii) *reference function* that connects the narrations to the fabula, allowing the derivation of the *plot*. In the fabula, each event is endowed with entities such as people, places, and physical objects.

Several recent models of narrative tend to be character-based, i.e. they view the narrative as a sequence of character actions that are related by causality links. As stated in Section 2.2, some ontologies, e.g. the Drammar ontology (Lombardo & Pizzo, 2014), also attempt at representing the characters' intentions and even their emotions.

Character-based models work especially well in the case of interactive narratives found in videogames or virtual environments, where characters are part of a relatively small, controlled world. In character-based models, the plot is dictated by the characters' goals, which cause them to perform actions and thus move the story forward.

However, when representing narratives set in the real world, the intentions of a single character often have limited relevance when compared to world-changing events that have a natural origin (e.g. the eruption of a volcano) or an artificial one (e.g. World War II). In this kind of narrative, which is the object of our study, *events* have the center stage.

What moves the story forward are then not the characters' goals, but rather the causality links between an event and another event. The plot is controlled by the narrator, who can choose which events to focus on and in which order they should be presented.

[1]In the present thesis we focus on narrations in natural language.

2.3.2 Ontologies for Representing Text

In order to understand how the representation of text can be improved in our ontology, we need to widen the field and look beyond ontologies of narrative, towards more general ontologies that focus on textual structure and semantics.

An investigation of these ontologies will help us answer research question Q1: "How can we model the textual and plot levels of narrative, and their relationship with the fabula, in a better way?".

Bibliographic Metadata

When modeling a textual object, a first level of representation is constituted by the description of the external features of the object, which may consist of metadata such as the title of the object, its author, its date of publication, etc. These metadata are often preserved in bibliographic records.

The de-facto standard ontology for representing bibliographic records is FRBRoo (Functional Requirements for Bibliographic Records, object-oriented) (Riva et al., 2008), a high-level conceptual model for bibliographic data. Compared to general metadata frameworks such as Dublin Core (Weibel et al., 1998), FRBRoo offers a much more thorough representation of the bibliographic data about a text. Furthermore, FRBRoo is an event-based ontology, and it is tightly integrated with the CIDOC CRM.

FRBRoo has recently been revised to create LRMoo (Library Reference Model, LRMoo) (Riva & Zumer, 2017). LRMoo attemps to streamline the representational complexity of FRBRoo, making it less cumbersome to adopt. Both FRBRoo and LRMoo are maintained by the International Federation of Library Associations and Institutions (IFLA) and are applied mostly in the Digital Library and Cultural Heritage fields.

Another ontology that allows the representation of bibliographic metadata is IAO (Information Artifact Ontology), a model of information entities that is based on the top-level Basic Formal Ontology (BFO) (Arp et al., 2015). IAO is maintained by the Ontology for Biomedical Investigations Consortium (OBI), and is adopted mostly in the biomedical field. A recent comparison between FRBRoo and IAO has been carried out by Garbacz (2016).

Textual Structure

The internal structure of a textual object can be described by looking at its subparts, e.g. a book may contain a set of chapters, and each chapter may contain a set of paragraphs. Different texts may have wildly different structures, for example, a poem is usually composed of lines instead of chapters or paragraphs. An ontology aiming to represent any possible textual structure will therefore have to be general in nature.

FRBRoo itself is more than adequate for representing general textual structures, allowing the modeling of part-of relations between each subpart of a textual object. FRBRoo opts not to offer specific classes to represent each possible subparts of a text (e.g. chapter, paragraph), keeping the model general enough to be applied to any kind of textual object.

A different approach is taken by DoCo (Document Components Ontology) (Constantin et al., 2016) and FaBiO (FRBR-aligned Bibliographic Ontology) (Peroni & Shotton, 2012), which define very narrow classes such as Paragraph, Abstract, or Preface. While this may be useful when representing homogeneous textual objects, it makes the model less flexible and harder to adapt to a general setting. In our case, we aim to make our ontology as general as possible, therefore adopting such approach would be too cumbersome.

Another important perspective that we need to consider is that of the Text Encoding Initiative (TEI) (TEI Consortium, 2021). TEI is a widely used standard for the representation of texts in digital form. While TEI is not an ontology, Ore and Eide (2009) have explored a possible integration between it and the CIDOC CRM. The latest version of TEI, called TEI P5, has moved towards an event-oriented model, more akin to that of the CIDOC CRM (Eide, 2014). Recent attempts to define a TEI ontology have been promising (Ciotti, 2018), but they are not yet advanced enough to be adopted for our purposes.

Recently, an extension of the CIDOC CRM, called CRMtex (Doerr et al., 2020), has been created for representing ancient texts. While this ontology defines some useful classes and properties, it is at an initial stage of development and is focused exclusively on ancient texts. For this reason, it would not be sensible to adopt it in our current work.

Among the aforementioned ontologies, FRBRoo (and its successor LRMoo) appears the more sensible choice for representing bibliographic metadata and textual structure in our work, for the following reasons:

- FRBRoo allows a thorough representation of both the bibliographic metadata and the internal structure of a textual object.
- The modeling of textual structure offered by FRBRoo is very general and not too cumbersome to use.
- FRBRoo follows an event-based approach, while other ontologies (such as IAO or FaBiO) do not.
- FRBRoo is tightly integrated with the CIDOC CRM, which is our main reference ontology.

Therefore, as will be discussed in Chapter 5, we are focusing on FRBRoo as a reference ontology for representing both bibliographic metadata and textual structure.

Linguistic Aspects and Textual Semantics

A further set of textual features that can be represented through an ontology model regards the linguistic aspects of the text, and also its semantics, i.e. the representation of the meaning of the text.

These topics are the realm of Linguistic Linked Open Data (LLOD) ontologies (Chiarcos et al., 2013). These ontologies, which are informed by standard practices in the field of Natural Language Processing, usually don't attempt to give an objective representation of the content of the text, but rather model a specific annotation of the text that describes one or more linguistic aspects of the text according to the interpretation of an annotator. A review of the main ontologies for LLOD was carried out by Bosque-Gil et al. (2018). Some of these ontologies focus exclusively on morphosyntactic aspects of the text, while others also allow the representation of textual semantics.

The most widely used ontology for linguistic annotation is NIF (NLP Interchange Format) (Hellmann et al., 2013), a standard that has been developed at the University of Leipzig since 2011. NIF is a very general ontology that allows the representation of many kinds of linguistic annotations, and is designed to be interoperable with several existing NLP annotation formats. For our purposes, NIF has the advantage of being widely adopted, and offering maximum compatibility with other models. However, one drawback of NIF is the fact that it does not define any classes or properties to represent the semantics of the text (e.g. that a fragment of text represents a person, or an event).

Based on NIF, the KYOTO project created KAF (KYOTO Annotation Format) (Bosma et al., 2009), an ontology-based semantic annotation format. The NewsReader project subsequently developed an updated version of KAF called NAF (NLP Annotation Format) (Fokkens et al., 2014). These ontologies add some more semantics to NIF, however they have not been adopted in a significant way outside of their respective projects.

An alternative to NIF is EARMARK (Extreme Annotational RDF Markup) (Barabucci et al., 2013), which aimed to allow the annotation of textual features that could be represented through previous annotation formats. Other relevant ontologies include ITS RDF[2], which is based on the Internationalized Tag Set standard (W3C MultilingualWeb-LT Working Group, 2013), TELIX (Rubiera et al., 2012), and GOLD (Farrar & Langendoen, 2003). However, these last ontologies are not widely adopted and do not appear to offer significant advantages over the ones based on NIF.

In addition, several ontologies have been created to support more specific kinds of linguistic annotation. For example, OLiA (Ontologies for Linguistic Annotation) (Chiarcos & Sukhareva, 2015) is a set of ontologies that support several types of morphosyntactic annotation. PROV-O (Lebo et al., 2013) allows the annotation of provenance information. At the current stage of our work, these seem overly specific for our purposes, but we may consider them for future extensions of our ontology.

Lexical annotation has also been investigated extensively, with the development of ontologies such as Lemon (J. McCrae et al., 2011), OntoLex (J. P. McCrae et al., 2017), PreMOn (Predicate Model for Ontologies) (Corcoglioniti et al., 2016), and OGL (Oxford Global Languages Ontology) (Parvizi et al., 2016). For the purposes of the present thesis, it is not strictly necessary to represent lemmas, but we are working an integration of our ontology with Lemon to support a concrete application in the context of the Hypermedia Dante Network project (see Section 8.2).

[2]http://www.w3.org/2005/11/its/rdf

A further subset of LLOD ontologies that is of great interest to us is the one that aims to represent the events that are described in the text. To the best of our knowledge, the only two ontologies that are specifically aimed at the annotation of textual events are GAF (Fokkens et al., 2014), a sibling of NAF developed in the NewsReader project, and CEVO (Comprehensive Event Ontology) (Shekarpour et al., 2019). Both can be easily integrated with NIF, but CEVO is more complex and also relies on ITS RDF, Open Annotation, and OLiA. As explained in Section 5.5.1, we have chosen to adopt GAF, which is simpler and more readily integrated with our ontology.

Finally, the W3C has standardized two ontologies for annotation on the Web: Open Annotation (OA) (Sanderson et al., 2013), and Web Annotation Data Model (WADM) (Sanderson et al., 2017). While these have sometimes been applied to the representation of linguistic aspects of a text, they are very general ontologies that do not offer any particular advantage for our purposes with respect to the LLOD ontologies discussed above.

Our implementation choices, based on a review of the aforementioned ontologies, are reported in Chapter 5, while in Chapter 8 we discuss the integration of our work with a lexical ontology to support a specific application.

2.4 Tools for Building Narratives

In the Digital Libraries field, the representation of narratives has been proposed as a functionality to improve discovery and exploration of the contents of DLs. In the following, we describe several projects that developed tools for building narratives, for different purposes. We took these tools into account in the development of our ontology and software.

It is important to note that these tools are not directly comparable to the one we propose in the present thesis, and to each other, because each of them follows a different model of narrative and was built for a different specific application. However, they can be useful as points of reference from which we can gather some useful experience, and draw some inspiration, in our work.

Several of the earlier systems for narrative construction were centered around museum collections. For example, Bletchley Park Text (Mulholland & Collins, 2002) allowed visitors to explore the collections by expressing their interests on some specific topics using SMS messages containing keywords. The semantic description of the resources was used to organize a collection into a personalized website based on the keywords chosen by the user. A similar later project was PATHS (Fernie et al., 2012), which allowed the creation of a personalized tour guide through existing Digital Library collections. The system allowed the definition of events linked to each other through semantic similarity relations.

Other tools focused more on the curation of narratives among museum objects. The Storyspace system (Wolff et al., 2012) allowed the description of stories based on events that span museum objects. The focus of the system was the creation of curatorial narratives from an exhibition. Each digital object has a linked creation event in the story of a heritage object. A similar system is Labyrinth 3D (Damiano et al., 2016), which integrates the semantic annotation of cultural objects with the interaction style of 3D games. The system allows the user to explore the collection using paths representing the semantic relations over cultural objects. Labyrinth 3D is based on the Drammar ontology model (Lombardo & Pizzo, 2014).

On the other hand, some projects focus more on the representation of historical events. CultureSampo (E. Hyvönen et al., 2007) is a portal and a publication channel for Finnish cultural heritage based on Semantic Web technologies. It uses an event-based model that allows the linking of events with digital objects, although its representation of events is more limited than that offered by our ontology. A related project, BiographySampo (E. Hyvönen et al., 2019), has developed a system to extract narratives from biographical dictionaries, represent them in a formal way using the CIDOC CRM and other ontologies, and publish them on the Web as Linked Data. The system has been used to build a portal containing more than 13,000 biographies of historical Finnish people. BiographySampo was the first large-scale project to adopt an automated narrative extraction system for populating an ontology-based model of narrative.

In the last few years, several projects have built tools to construct narratives on top of the collections of the Europeana digital library (Europeana, 2014) to construct narratives and facilitate the discovery and exploration of their digital objects. For example, EAGLE (Mannocci et al., 2014) is a European project (2013-2016), which was part of Europeana. The aim of the project was to build a reference portal for the epigraphic images and texts of the Ancient World. EAGLE enriched Europeana with many digital objects, and developed a storytelling application which allows teachers and experts to assemble epigraphy-based narratives. Historiana[3] is an online educational multimedia platform that offers historical content, learning activities, and digital tools for exploring events from European history. To enrich its contents, Historiana imports digital objects from Europeana.

Other projects allow the visualization of data on a particular topic contained in existing knowledge bases in the form of narratives. For example, Thinkpedia (Hirsch et al., 2009) is an application which produces visualizations of the semantic knowledge contained in Wikipedia, allowing the user to explore the semantic graphs of the two knowledge bases in an accessible and interactive way. Histography[4] is a Web application that visualises events imported from Wikipedia, spanning the entire history of the universe. The user can focus on a particular period of history or even a specific event. These tools have the disadvantage of not allowing the user to alter the knowledge presented to them, therefore they cannot be considered tools for creating narratives.

On the other hand, Histropedia[5] allows users to create or view timelines on topics of their choice by importing statements from Wikidata. Links to related Wikipedia articles and Wikimedia Commons images are automatically added, resulting in rich spatio-temporal visualizations. The scope of the project spans research, education, and tourism (Mietchen et al., 2015). Similarly, Chronas[6]

[3]http://historiana.eu
[4]http://histography.io
[5]http://histropedia.com
[6]http://chronas.org

is a chronological and cartographical history application, with a special focus on visualising maps of the world across human history. The knowledge visualised in the application is automatically imported from Wikipedia, but manual additions and modifications by the users are allowed. Storify[7] was a free storytelling environment allowing users to aggregate social media posts, Web pages, photos, and videos, to curate narratives about news events. The tool was also applied for educational purposes (Cohen & Mihailidis, 2012). These tools are more user-centered, allowing the construction of narratives based on existing data. However, none of them are based on an ontology of narrative.

2.5 Narrative Extraction from Text

In our research, narrative extraction techniques can be applied to the investigation of research question Q2, i.e. "How can we extract elements of narrative from text?". In particular, we want to understand how much of the narrative structure can be extracted automatically, and with what level of accuracy.

In the 1990s, the input for a renewed interested in the field of narrative extraction from text came from Natural Language Processing researchers, who started studying the automatic extraction of events, first through rule-based systems (Chinchor, 1998; Lehnert et al., 1992) and later through supervised classifiers (Chieu & Ng, 2002; Freitag, 1998). The accuracy of these initial systems was limited.

Starting in 2004, the ACE program (Doddington et al., 2004) has focused on extraction of entities, relations, and events from natural language text. In particular, the ACE 2005 competition involved the extraction of several classes of events from an annotated corpus of newspaper articles. This is still the gold standard against which event extraction algorithms are evaluated. A helpful comparison between the main annotation tasks of ACE 2005 is provided by Aguilar et al. (2014).

Such efforts were soon applied to computational narratology. Chambers and Jurafsky (2008) were the first to apply event extraction algorithms to narrative extraction from text, and led the way in the development of unsupervised approaches for this task (Chambers & Jurafsky, 2009). Their approach relied heavily on the concept of "narrative chains" that had to be identified in the text. Around the same time, Elson et al. (2010) extracted social networks from a set of 19th century literary texts, showing that it was possible to build such kind of network through an automatic extraction of relations from text.

Narratives were later applied to information extraction from news streams (Vossen et al., 2015) in the context of the NewsReader project (Vossen et al., 2014). This project had the advantage of adopting a linguistic ontology to represent the events described in news articles. At the same time, significant advances were accomplished in the field of temporal information retrieval. For example, HeidelTime (Strötgen & Gertz, 2010) is able to recognize temporal mentions in the text and also normalize them to obtain their datetime value. We refer the

[7] https://storify.com

interested reader to the comprehensive review by Campos et al. (2014) for more information about this topic.

Some later works, including Linger and Hajaiej (2020) and Piskorski et al. (2020), also adopted narratives as a means to extract the information found in news articles, while Hausner et al. (2020) applied narratives to knowledge discovery in repositories of legal documents.

In the following two sections, we focus on two specific aspects of narrative extraction from text that are particularly relevant to our present work: event extraction and entity linking.

2.5.1 Event Extraction

Event extraction is the task of extracting the events described in the text, including both the event mentions and the entity mentions that act as arguments of those events (e.g. people, places, temporal knowledge, etc.).

Event extraction techniques can be applied to the investigation of research question Q2, i.e. "How can we extract elements of narrative from text?", because events are the most important element in our model of narrative.

Since the 2010s, research in event extraction as a classification task has made significant progress, with the development of new feature-based and pattern-based approaches (Cao, 2017) and a gradual shift from traditional dependency parsing (Li et al., 2013) to deep learning (LeCun et al., 2015).

In particular, convolutional neural networks are often able to identify the structural features of a sentence in a deeper way than feature-based approaches (T. H. Nguyen & Grishman, 2015), but they also have shortcomings, e.g. they find difficulty with sentences containing multiple events (Chen et al., 2015). Recurrent neural networks, and in particular bidirectional LSTMs (Schuster & Paliwal, 1997), are better able to capture long-term dependencies in natural language text (Feng et al., 2018; T. H. Nguyen et al., 2016).

More recently, the newer Transformer architecture (Vaswani et al., 2017), including pretrained models and embeddings based on the BERT language representation model (Devlin et al., 2018), have been applied successfully to the task of event extraction. For example, DyGIE++ (Wadden et al., 2019) relies on BERT to jointly extract entities, events, arguments, and relations from text. This software is able to extract both events and arguments in a single neural network model, with better accuracy than previous systems.

Yang et al. (2019) also explored pretrained neural network models for both event extraction and generation, while Mellace, Antonucci, et al. (2020) applied a deep learning approach based on BERT to the modeling of character relationships extracted from narrative texts. From these studies, it is clear that the application of large linguistic models is currently the most accurate way to extract narrative events from text. For this reason, we have chosen to investigate the application of BERT in our work (see Section 6.6.1).

It should be noted that other, more novel approaches exist that view event extraction as a different task than classification. For example, Liu et al. (2020) model event extraction as a machine reading comprehension task, while X. D. Wang et al. (2020) model it as a question answering problem. These newer approaches are promising, but since they are at an early stage of development we will not consider them for the purposes of the present thesis.

In addition, research has been very active on the extraction of events from social media streams (Becker et al., 2011; Sakaki et al., 2010; Weng & Lee, 2011) and the subsequent construction of a knowledge graph based on the detected events (Shekarpour et al., 2018). Event extraction has also been studied extensively in the biomedical field (Björne & Salakoski, 2011; Yakushiji et al., 2000).

These works are not particularly relevant for the kind of narrative extraction that we are aiming for in the present thesis, but some of the approaches contained in them may be useful to us in the future, if we choose to expand our research towards those directions.

2.5.2 Entity Linking

Entity linking is the task of linking entity mentions found in a natural language text to the corresponding entities in a knowledge base (Shen et al., 2015). In the context of narrative extraction, entity linking can be applied to connect entity mentions (including event mentions) to their corresponding representations in a suitable knowledge base (such as Wikidata (Vrandei & Krötzsch, 2014)).

By performing entity linking, each narrative graph becomes linked to a larger graph, instead of being isolated. When representing narratives set in history, as we do, performing entity linking means being able to import external data, images, and other media that would otherwise have to be manually attached to the narrative by a human user.

Entity linking techniques can be applied to the investigation of research question Q2, i.e. "How can we extract elements of narrative from text?". In particular, entity linking can improve the extracted narrative graph by identifying the semantic value of entity mentions found in the text.

Most of the current entity linkers adopt a single language edition of Wikipedia[8] as reference knowledge base. For instance, TagMe (Ferragina & Scaiella, 2010) and its successor WAT (Piccinno & Ferragina, 2014) use the English Wikipedia graph, made of links between pages, to associate named entities in a text to the corresponding pages about them found in Wikipedia.

This reliance on a specific language edition of Wikipedia represents a significant limitation, because the set of possible matches depends on the number of entities described in that specific edition. Using a wider language-agnostic knowledge base would be a significant improvement, but the hurdle is that larger knowledge bases often have much shorter textual descriptions of the entities. Until a few years ago, no entity linker had successfully solved this issue. Even the ones that were able to link to a multilingual knowledge base such as Wikidata, e.g. the one by Foppiano and Romary (2018), primarily targeted Wikipedia, and only then they linked the results to Wikidata.

However, in the last few years, several entity linkers have been developed that are able to link to a multilingual knowledge base. In particular, OpenTapioca

[8]https://wikipedia.org

(Delpeuch, 2019) is a lightweight entity linker based on topic similarity that is synchronized in real time with Wikidata. Falcon 2.0 is a rule-based the first joint entity and relation linker based in Wikidata (Sakor et al., 2020). PNEL (Banerjee et al., 2020) is an entity linker based on the concept of Pointer Networks, and relies on pre-indexed embeddings and entity labels as features for retrieval. The entity linker by Botha et al. (2020) also returns links to Wikidata, however it has the drawback of relying on a small multilingual corpus from WikiNews to select the candidate entities.

Among the entity linkers mentioned above, OpenTapioca looks particularly interesting for our purposes, because it is very flexible, allowing the user to use SPARQL queries to select the reference graph, it doesn't require a custom training corpus, it is very easy to integrate with existing software, and it is available under an open license (see Section 6.6.5).

This concludes the chapter about the state of the art. In the following chapter, we will describe our methodology and our requirements for the development of both the Narrative Ontology and the NarraNext system for narrative extraction from natural language text.

CHAPTER 3

Methodology

This chapter presents the methodology that we followed to develop the ontology and the tool for narrative extraction from text. First of all, we present the methodological approach that we followed throughout our research work. Then, we report our research questions and discuss how we plan to answer them. Finally, we report a set of requirements that we have identified based on the research questions.

3.1 Methodological Approach

The methodological approach that we followed in our research work to introduce narratives in Digital Libraries is very similar to the one that characterizes a common workflow to develop an algorithm in Computer Science (Levitin, 2007), that is composed of the following phases:

1. Formalization of the problem
2. Computational analysis
3. Development of a new algorithm
4. Experimentation with a case study
5. Evaluation

In our research, we decided to adopt a similar approach, adapting the phases of algorithm development to our goals.

In particular, our methodological approach consists of the following four phases:

1. Creation of a conceptualization of the domain, in which the issue is described and analysed in its main parts.
2. Development of an ontology as the specification of the conceptualization in terms of a logical theory whose axioms admit as models those licensed by the conceptualization.
3. Implementation of a tool to populate the ontology starting from natural language text.
4. Experimental validation of the ontology and evaluation of the results.

In the present thesis we cover all the phases reported above. Our conceptualization is described in chapter 4, and its specification through an ontology is reported in chapter 5. The implementation phase is described in chapter 6, while the validation of our work is reported in chapter 7.

Concerning evaluation, which is the last phase of the methodology, we note that our ontology is endowed with a reasoning algorithm that is sound, complete, and efficient, as just explained.

We consider this fact as a qualitative validation of our work, and a crucial one: without such a reasoning algorithm, our ontology would not be usable, given that we aim at *computable* narratives. Of course, a theoretical validation is not sufficient, it must be complemented by a pragmatic one.

In this respect, it should be noted that the very notion of narrative has been recognized as a complex one, and that narratologists have not yet reached a commonly-shared understanding of the fundamental aspects of narrative (Herman, 2009). Under these circumstances, no ontology can be validated as *the* ontology of narrative.

Pragmatically, then, all that can be done is to test a tentative ontology that is promising, to see whether it responds reasonably well to the requirements at hand. This is what we have done with our ontology, by applying it in two evaluation contexts: one through a validation experiment, and the other on the field in the context of the Mingei and Hypermedia Dante Network projects.

3.2 Research Questions

The general research questions that we have identified at the beginning of our study are as follows:

Q1. How can we model the textual and plot levels of narrative, and their relationship with the fabula, in a better way?

Q2. How can we extract elements of narrative from text?

Q3. How can we validate this work, and which innovative applications can we develop based on it?

These research questions can be further specified as follows:

Q1.1 How can we model the narration (textual level of the narrative) in a more effective way?

Q1.2 How can we model the plot of the narrative in a more effective way?

Q1.3 How can we model the relationship between fabula, plot, and narration in a more effective way?

Q2.1 What are our requirements for narrative extraction from text?

Q2.2 What is a viable architecture for narrative extraction from text?

Q2.3 How can we implement the components of this architecture?

Q2.4 How can we evaluate the effectiveness of these components?

Q3.1 How can we develop a narrative building tool that is able to extract narratives from text?

Q3.2 How can we build an effective user interface for this tool?

Q3.3 How can we apply the ontology in a real-world setting?

In the present thesis we will do our best to answer these research questions, by working on the following general answers:

A1. A new version of the Narrative Ontology that is better able to represent the textual and plot levels of narrative, including both structural and semantic aspects of natural language text; this answers research question Q1 by providing solutions for both shortcomings that we had identified in our previous computational representation of narrative.

A2. A narrative extraction system that is able to detect narrative elements in natural language text and allows a human user to build a narrative based on them; this answers research question Q2 by providing NarraNext, a comprehensive tool based on several integrated techniques to extract narratives from text in a semi-automatic way.

A3. An evaluation on a set of biographies that validates our narrative extraction system, and in addition two further applications of our ontology in the context of Cultural Heritage and Digital Libraries; this answers question Q3 by verifying the effectiveness of the NarraNext tool and building concrete applications based on the Narrative Ontology.

These developments will be presented in the following chapters. But before that, we have identified some requirements that we aim to address in our research work.

3.3 Requirements

In this section we present the requirements that we aim to satisfy in the development of the Narrative Ontology and the NarraNext tool for narrative extraction.

In general, an information system serves the needs of a community of users, therefore analysing users' requirements is an integral part of information systems design and it is important for the success of interactive systems.

As specified in the ISO 9241-210 standard[1], understanding the needs and requirements of the users is the first step to develop successful systems and products.

Indeed, the result of this analysis can bring a project an increase of productivity, a better quality of the work, smaller costs for providing support and training, and improvement of users' satisfaction (Doll & Torkzadeh, 1988).

The first step in users' requirements analysis is to collect background information about the users and the processes that currently take place, through structured interviews.

In our project, we used this approach to identify a set of requirements on the easy and user-friendly creation and visualization of narratives, and on the knowledge required to carry out such tasks.

The preferred users for this research are scholars who want to create and access narratives about the life and the works of the authors they study. These scholars naturally refer to DLs for their work: they use the search service of the DL to discover and access the relevant resources; they use the tools made available by the DL to carry out their work, including communicating with other scholars; finally, they make use of the DL to disseminate and preserve the result of their work.

Therefore, our close relationship with these scholars has been highly beneficial for the definition of our requirements.

3.3.1 Identifying Users for Narratives

In this section, we present some use cases of how NarraNext could be used by the kinds of users reported below. Through the tool, these users will be able create and visualize their own narratives or access the narratives built by somebody else.

Scholar A scholar who is studying the life of a historical figure may use the tool to annotate the life of that person according to the available biographies, in order to compare the different narratives that exist about a person's life.

High School Professor NarraNext may be used by a professor as a learning tool. The professor may create a narrative on a topic of study, by importing it from a natural language text, enrich it with some digital objects, and then show it to the students through a timeline visualization.

[1]https://www.iso.org/standard/52075.html

Student A student may be interested in using the tool as a study aid, for reviewing a text about a specific subject of study and memorizing the information by creating a digital narrative about it, which could then be used to review the topic at a later date.

Museum Curator A museum curator could use NarraNext while planning an exhibition about a specific artist, by linking the works of the artist to biographical events in order to give the visitors an overview of the life and works of the artist. The final interactive narrative may be made available as part of the exhibition.

Biographer NarraNext may be used by a biographer to produce a digital version of one of the biographies they wrote. The narrative will then become accessible on the Web.

3.3.2 Requirements for the Narrative Ontology

The ontology should be able to represent the following aspects of a narrative:

1. representing the *fabula* of the narrative
2. representing each *event* of the fabula, including its time, place, and any related entity
3. representing *relations* among events
4. representing the *plot* of the narrative
5. representing the *narration*, including the textual structure and semantics
6. allow for multiple narratives to coexist in the same DL

3.3.3 Requirements for the NarraNext Tool

This section reports the requirements for the development of the NarraNext tool. We have identified three sets of requirement: (i) for narrative extraction, (ii) for the interface of the tool, (iii) for the final visualizations of the narrative.

Requirements for Narrative Extraction

The requirements that we have identified for narrative extraction are as follows:

1. *Detecting events in the text*. First of all, events have to be detected in the text. For example, the text "Dante died" expresses the event "Death of Dante".
2. *Classifying events*. Events that are detected in the text need to be classified, i.e. categorized into a set of event classes (for example, the 100 Years War could be classified in a class called *Conflict*).
3. *Detecting event arguments*. We need to extract from the text the entity mentions that act as event arguments, i.e. identify the space, time, and the entities that took part in the event (people, objects, concepts, etc.).

4. *Detecting relations between arguments and events*. We need to identify which entities act as arguments of which events, and if possible classify this relation.

5. *Detecting temporal entities*. Temporal entities such as dates, years, and other indications of time need to be extracted from natural language text.

6. *Normalizing temporal entities*. Temporal entities have to be normalized, i.e. their value needs to be understood to correctly place each event in the narrative's timeline.

7. *Making sense of coreferences found in the text*. It would be useful to understand when the text references an entity without mentioning it explicitly, e.g. through a pronoun.

8. *Linking entities to a reference knowledge base*. It is useful to perform entity linking and event linking to connect them to an external knowledge base.

9. *Linking events to a reference knowledge base*. Similarly to entities, events could also be linked to an existing knowledge base.

10. *Constructing the narrative*. Once we have identified all entities and all events, and linked them to each other, we have to construct the fabula and plot of the narrative.

Requirements for the User Interface

The requirements for interface of the tool are as follows:

1. Creating events and placing them on a timeline

2. Setting the title of each event

3. Setting the time of each event

4. Setting the place of each event

5. Linking to each event the entities that participate in it (people, objects, concepts, etc.)

6. Importing such entities from a reference knowledge base, or defining new ones if they cannot be found in the knowledge base

7. Defining the type of each event choosing from a list of pre-defined options.

8. Re-ordering the events based on their position in the plot

9. Linking one or more digital objects to an event (e.g. objects collected in a DL)

Requirements for Visualizations

In addition to the creation of the narrative, the tool should allow the user to visualize the narrative and download it for further processing or sharing it on the Web. Concerning visualization, the tool should allow:

1. Visualizing a narrative on an interactive timeline.
2. Visualizing all events in a narrative, or a subset of them defined by the final user, in a tabular form, along with their primary sources. The table should be exportable in CSV format.
3. Visualizing events that happened in a user-specified range of time, also in form of table, exportable in CSV format.
4. Visualizing a specific event, chosen by the final user, and its related entities as a network graph.
5. Visualizing a specific entity, chosen by the final user, and its related events as a network graph.

This concludes the chapter about methodology. In the following chapter, we will present the conceptualization of narrative that is at the basis of our Narrative Ontology.

CHAPTER 4

A Conceptualization of Narrative

As explained in Chapter 3, the first step of our methodology is the creation of a conceptualization of the domain of knowledge that we aim to represent, i.e. the concept of narrative, its components, and its constituent levels (fabula, plot, narration). This chapter presents the conceptualization we developed. This conceptualization will then be expressed in a more formal way in Chapter 5, both through a specification in First Order Logic and through a more concrete expression based on the languages of the Semantic Web.

4.1 Background

As we stated in Section 2.1, the study of narrative goes back to Aristotle (Aristotle, 2013) and to the fourth century BC, and has been further elaborated by many philosophers afterwards.

In the 1920s, the Russian formalists offered an account of narratives that has been used for a systematic study of narrative structure (Shklovsky, 1997). This account has eventually gave rise to narratology as an autonomous scientific discipline.

According to the Russian formalists, a narrative consists of three main elements:

- the *fabula*, i.e., the story itself as it happened, in reality or in fiction;
- the *narration(s)*, i.e., one or more expressions, each in its own language and medium, that narrate the fabula. Each narration corresponds to Bal's definition of *presentation* (Bal, 1997);

- the *plot*, i.e., the story as it is narrated by the narrator. The plot corresponds to the *syuzhet* of the Russian formalists and to Aristotle's *logos.*

Current DLs contain only the *narration* level of the narrative, i.e. the expression of the narrative through a media object. To enhance the representation of narratives in DLs, we propose adding a formal expression of the *fabula* and *plot* levels of the narrative.

4.2 Conceptualization

This Section presents our view of a computable representation of narrative, as informed by the background reported in Chapter 2. We introduce the relevant notions both at an informal level and more formally in set-theoretic terms. An initial version of this conceptualization has been reported in (Bartalesi et al., 2016). The present version extends the initial one in several important ways.

4.2.1 Narrative

We view a narrative as a story told by a narrator, which may be an individual person or a group of persons taking up the role of the narrator. Each narrative reflects the point of view of its narrator. The stories in the scope of our work are generally real stories of the present or the past.

Fictional stories may also be expressed in our ontology. However, since it is more important for us to support science rather than fiction, these stories have to be consistent with the axioms on physical reality that our ontology is able to capture. This excludes stories in which, for instance, effects precede causes, events nest circularly, or objects bilocate.

In our conceptualization, a narrative consists of three main elements:

1. the *fabula,* i.e., the story itself as it happened, in reality or in fiction;
2. the *narrations,* i.e., one or more expressions, each in its own language and *medium,* that narrate the fabula. Each narration correspond to Bal's definition of *presentation* (Bal, 1997);
3. the *reference,* i.e., a relation that connects (fragments of) the narrations to (fragments of) the fabula

By combining the *fabula* with the *narration*, it is possible to derive of the *plot* of the narrative (also called *syuzhet* by Russian Formalists).

4.2.2 Fabula

A fabula consists of events, each of which encompasses a significant fragment of the story. We define an event as a group of coherent phenomena situated in space and time.

An event is contextualized in terms of the entities that participate in it. In addition to space and time, the other entities that participate in the event can be identified as having persistent characteristics of structural nature. They may be physical entities (e.g. people, physical objects) or conceptual entities (e.g. concepts, ideas). Actions, and more generally activities, are special cases of events. In the fabula, the events are ordered chronologically, as defined by Russian formalism.

Relations Among Events

We have identified three main relations that may exist among the events of the fabula: the *mereology* relation, the *temporal occurrence* relation, and the *causal dependency* relation.

The *mereology* (or *event composition*) relation connects events to other events that include them as parts, e.g. the birth of a person is part of the broader event of the life of that person.

The mereology relation is a strict partial order, i.e. it is an irreflexive and transitive relation over the fabula's events; consequently, it is asymmetric, and more generally acyclic, so that no event is a sub- or super-event of itself or some other event.

The *temporal occurrence* relation associates each event with a time interval during which the event occurs. As such, the temporal occurrence relation is a total function. In turn, time intervals are connected to each other through the 13 relations of Allen's interval algebra (Allen, 1983). These relations are jointly exhaustive and mutually exclusive, therefore each pair of events is connected by one and only one Allen relation. Each time interval has exactly one starting time point and one ending time point. Time points are connected to each other through before, after, or equals relations;

The *causal dependency* relation relates pairs of events such that the occurrence of the former causes the occurrence of latter, e.g. the eruption of the Vesuvius caused the destruction of Pompeii. Clearly, a formal account of the causal dependency relation requires a complete knowledge of the laws governing reality, and is therefore out of question. Instead, we will confine ourselves to assert that causal dependency is a *strict partial order*. In this case, acyclicity guarantees that no event is at the same time cause and effect of itself or some other event.

It is important to note that we are not interested in modeling the mechanical causal relationships that connect events in a physical or chemical process. Such laws are partially known. Rather, we are interested in a more generic notion of causality, whereby the connected events may be years (or centuries) apart in time, and the causal connection between them may be indirectly established through other events, which may or may not be represented in the fabula. In our conceptualization, each causality link among the events of the narrative is always established by the narrator, who has the ultimate responsibility to decide if and when two events are causally connected in their own opinion.

Causal dependency is determined by the laws of nature that are in force in the reality where the fabula is located. Such laws are only partially known; for instance, the laws governing the behaviour of people, assuming they exist, have not yet been formalized at the level of precision at which some of the laws of physics have; moreover, many laws operate at a level of granularity that may, and in general will not be the same as the level of the fabula. As a consequence, it is to be expected that causal dependency will only be partially modelled in the ontology that specifies the present conceptualization.

In addition to the features of the individual relations in a fabula stated so far, the following conditions are met by every fabula:

1. The period of occurrence of an event is included in the period of occurrence of any of its super-events.

2. The beginning of occurrence of an event precedes the beginning of occurrence of any event that causally depends on it.

The expression of the inclusion and precedence relations among time intervals will be dealt with in Section 5.2.3, upon considering the representation of temporal knowledge in narratives.

4.2.3 Narrations

In a narrative, a fabula may have any number of narrations, each of which has the obvious characteristic of being *about* the fabula. Intuitively, this aboutness is a notion of *representation* between fabula and narration, in the sense that any narration of the fabula must somehow represent the fabula, in whole or in part. Logically, this amounts to say that any proposition in the narration, whether explicitly or implicitly stated, must be true in the fabula.

Each narration has one or more narrators (the authors of the narration), and of a *narration content*. In general, the narration content is a message, and it may take any form in which a fabula can be communicated, ranging from text, to audio-visual message, to theatrical enactment, etc.

For obvious reasons, we are interested in narrations that have at least one digital representation, whether such representation is only the carrier of a non-digital narration (e.g. an audio-visual recording of a theatrical piece) or a born-digital narration (e.g. a born-digital text or a video game). In our conceptualization of a narration, the content will therefore be any media object, i.e. a text, an image, an audio-visual object, or any complex multimedia object that a particular narrator, or group of narrators, choose to tell their version of the fabula.

In order to exemplify the structure of the narration, in the rest of this Section we characterize the structure of a narration having a digital *textual content*. The structure of a text can be derived through its decomposition in *textual units* (e.g., chapters, paragraphs), as established by the narrator(s). Each textual unit is linked to its containing unit, and ultimately to the whole text, through a *textual mereology* relation.

More technically, a narration n has:

- a content, that is a finite length string s_n over some alphabet; the relation between a narration and its content is a function;

- a structure, given by a set of units, each of which is a sub-string a of s_n identified by its boundaries l_a and r_a; clearly, $1 \leq l_a < r_a \leq |s_n|$, for each unit a.

The set of units in a narration structure satisfies an obvious containment criterion, which can be characterized by a positive integer k_n, the number of levels of n, and a function f from $\{1, \ldots, k_n\}$ to sub-intervals of $[1, |s_n|]$, giving the units of each level in terms of their boundaries. The structural units at level i are obtained by dividing the units at level $i-1$ in disjoint portions. These portions correspond to the mathematical notion of partition, which we apply to the boundaries defining each unit. In particular:

1. $f(1) = \{[1, |s_n|]\}$, i.e., the first level has only one unit, consisting of the whole narration content, having as boundaries the boundaries of the whole narration content;

2. $f(i+1) = \cup_{u \in f(i)} \pi(u)$ for $1 < i \leq k_n$, i.e., the boundaries of the units at the $(i+1)$-th level are all partitions $\pi(u)$ of the boundaries of every unit u at level i.

To exemplify, a narration can be structured as a book divided in chapters, each divided in sections. The levels of this structure are three. The first level consists of the whole book, as a single block of text given by the string s_n; the boundaries of this unit are therefore 1 and $|s_n|$. At the second level, the book unit of the first level is divided into sub-units, each being a chapter. At the third level, each chapter of the second level is further partitioned into sub-sub-units, each being a section of the partitioned chapter. Clearly, not all units need to be partitioned; if a unit u is not partitioned, i.e., a chapter has no section, then $\pi(u) = \{u\}$.

The mereological relation between the units of the narration structure reduces to containment between the relative intervals, i.e., unit a is part of unit b iff the boundaries of a define a sub-interval of the interval defined by the boundaries of b. It is not difficult to see that such relation is a partial order that can be made strict by removing the reflexive pairs.

Reference The *reference* is a relation that connects regions of narrations, which we call *narrative fragments* (or simply *fragments*), to events of the fabula. Each fragment is maximal, in that it comprises all portions of the narration that narrate the same event.

A fragment is identified in ways that depend on the structure of the narration. For instance, a textual fragment will be a set of disjoint intervals, each giving the boundaries of texts narrating the same event. A fragment that narrates an event necessarily narrates any of its super-events, and no other event.

Using the reference relation, it is possible to reconstruct the plot of the narrative, i.e. the sequence of fragments in the order established in the narration by the narrator.

Because a fabula is identified by its composing events, two narrations of the same fabula may differ for any combination of the following:

1. the set of events of the fabula narrated by the narrations; each narration may pick a different subset of events, as a way of giving more emphasis to certain aspects of the story;

2. the order in which the selected events are narrated;

3. the expressions used for the narration.

Two narrations offering accounts of the same story that are incompatible, in the sense expressed above, are not narrations of the same fabula. This fact does not prevent comparing the narrations, for instance to appreciate their differences.

Plot We define the *plot* as the order of events as told by the narrator. It should be noted that each narration has exactly one plot, but each plot can have one or more narrations.

The plot can usually be inferred through the reference relation, therefore it would be redundant to represent it explicitly in NOnt. However, there may be some narratives in which the narration level is not explicitly defined, or is unknown. In Section 5.5.2 we discuss this issue and propose a possible solution as an extension of NOnt.

4.2.4 Representing Narratives in DLs

In our view, a Digital Library (DL) should provide digital representations of narratives as first-class citizens of the DL. For simplicity, we will call such digital representations *narratives* whenever no ambiguity can arise.

For completeness, the narratives in a DL should encompass all aspects discussed in the previous Section, i.e. narrations, fabulae and reference relations. While it is expected that a DL already possesses narrations in digital form, our work is motivated by the target of lifting such narrations into narratives, endowing them with a formal representation of the corresponding fabulae, acting as a semantical counterpart of those narrations.

This two-level representation of the narrative allows supporting the union of the use cases supported by the purely syntactical (i.e. based solely on narrations) and the purely semantical (i.e. based solely on fabulae) representations. From now on, when there is no ambiguity, we will speak of the fabula of a narrative meaning the representation of the fabula, as we do for narratives.

A narrative may be constructed in at least two different ways:

- starting from a narration and associating it to a fabula, or
- starting from a fabula and associating it to a narration of the fabula.

In the former case, the involved process is *formalization:* the narration is decomposed into meaningful events and each event is formally represented via statements drawn from the narration; the reference relation is used to establish the proper connection between fragments of the narration and the corresponding formalizing events.

In the latter case, the involved process is *documentation:* the events of the fabula are given, and the narrative is constructed by linking each of them to a narration fragment that illustrates the event, using for that purpose (the inverse of) the reference relation. In either case, automatic or semi-automatic methods can be devised to support the process and make it scale.

It must be noted that either the narration or the fabula of a narrative may provide an incomplete, or even inaccurate, account of the story that the narrative is about. In each of them, events may be reported by omitting or mistaking their temporal or spatial occurrence; likewise, the participation of persons in events or the causal dependencies between events may be omitted or mistaken. For this reason, the fabula of a narrative must be treated as a knowledge base (KB for short), i.e. a set of statements giving the best available approximation of the fabula according to the narrator of the narrative.

The relationship between the real fabula and its representation may be precisely characterized from a logical point of view as follows. A real fabula f may be seen as a set of possible worlds, namely of the worlds that are compatible with the events in the fabula and the relationships that link these events to each other and to their factual components.

Let S_f be the maximal set of formal fabula statements that are true in every world in f. A language for expressing these statements will be introduced in the next Section, but for now it suffices to assume that such language exists. Let k be a non-empty KB with the formal representation of f. Then,

- k is an *accurate* representation of f iff every statement in k is true in the fabula, formally iff $k \models S_f$, where $\models$ is the logical implication relation.

- k is a *complete* representation of f iff k says everything about f, formally iff $S_f \models k$.

Accurate and complete accounts of the fabula are therefore knowledge bases k that are equivalent to S_f,. Needless to say, such accurate and complete accounts are idealizations that real representations can only try to approximate.

As a consequence of the inaccuracy or incompleteness of fabulae, and therefore of narratives in general, it may be the case that two narratives provide different versions of the same story, making different statements about the same events, possibly leading to contradiction. For instance, a narrative about the life of Dante Alighieri may include a travel to France as an event, while another narrative may deny the occurrence of that event, e.g. by placing Dante at a different location at the same time.

Actually, the presence of different versions of the same story is not to be seen as accidental or undesirable in a DL. To the contrary, it manifests different points of view that it is important (in some cases vital) to document. On the other hand, the arising of logical contradictions in a KB is highly undesirable, because it makes the KB unusable: since everything logically follows from an inconsistent KB, the answers to queries performed against an inconsistent KB will not be reliable.

In order to enable a DL to hold incompatible narratives while at the same time avoiding the rise of inconsistencies, we view each narrative as a separate KB, and a DL as a set of narratives, possibly sharing a common set of factual components that occur in the fabulae of these narratives.

This concludes our conceptualization of narrative. In the next chapter, we will provide a formal specification of this conceptualization, taking the form of a narrative expressed in First Order Logic (Smullyan, 1968). Then, we will express the ontology through the OWL (W3C OWL Working Group, 2012) and SWRL (Horrocks et al., 2004) languages of the Semantic Web.

CHAPTER 5

The Narrative Ontology

In this chapter we present an ontology of narratives, called Narrative Ontology (NOnt for short), which specifies the conceptualization given in the previous chapter. We will describe the formal expression of the ontology through First-Order Logic (Smullyan, 1968), and its concrete representation through the languages of the Semantic Web.

5.1 Outline

The ontology that is presented in this chapter has been recently described in (Meghini et al., 2021). This ontology represents a significant extension of the one reported in a previous study (Bartalesi, Meghini, & Metilli, 2017), with many changes and improvements. The most significant ones are as follows:

- The ontology has been redesigned and fully expressed in First-Order Logic (Smullyan, 1968) (Section 5.2);
- The ontology has been updated with better support for multiple ontologies stored in a single Digital Library (Section 5.2.2);
- The ontology has been mapped with three reference ontologies: the CIDOC CRM (Doerr, 2003), FRBRoo (Riva & Zumer, 2017), which supports the representation of textual structure, and also OWL Time (Hobbs & Pan, 2020) for temporal representation (Section 5.4);
- The ontology has been updated to support temporal reasoning on the narrative, and a set of SWRL (Horrocks et al., 2004) rules has been created for this task (Section 5.2.3).

Furthermore, at the end of this chapter we propose the following further additions to the ontology:

- an integration with the NIF (Hellmann et al., 2013) and GAF (Fokkens et al., 2014) ontologies, to support the representation of textual semantics and make the ontology compatible with Linguistic Linked Open Data (LLOD) standards (Section 5.5.1);
- an explicit expression of the *plot* level of the narrative (Section 5.5.2);
- a representation of event subclasses through a mapping with the CIDOC CRM and Wikidata (Section 5.5.3);
- a representation of event schemas, i.e. descriptions that give the structure of whole classes of events (Section 5.5.4).

5.2 Expression in First-Order Logic

The ontology is expressed through the language of First-Order Logic (Smullyan, 1968). This choice allows us to represent the ontology through a formal language that is independent of any specific concrete expression. Furthermore, the choice of FOL guarantees maximum expressivity.

Due to the fact that a DL includes a *global* KB, i.e. a set of statements that document the narratives encompassed in the DL, NOnt will be split in two parts: (i) NOntNar, including the classes, properties and axioms for expressing individual narratives, and (ii) NOntDL, including the classes, properties and axioms for expressing the knowledge in the global KB of a DL.

Before delving into the definition of the ontology, we will now discuss some epistemic aspects at the basis of NOnt.

5.2.1 The $\mathcal{L}_n$ Language

Our task requires the identification of a specific first-order language $\mathcal{L}_n$ that is able to capture the intended meaning of our ontology for narratives. $\mathcal{L}_n$ is derived from the $\mathcal{L}$ presented in (Levesque & Lakemeyer, 2001), and it includes the sentences that are required to axiomatize the narratives.

As customary in logic, the alphabet of $\mathcal{L}_n$ includes two kinds of symbols: logical and non-logical. Logical symbols are the symbols whose usage and interpretation are fixed, while non-logical symbols are the symbols that are domain-dependent. The logical symbols of $\mathcal{L}_n$ are:

- countably many variables $x, y, z \ldots$;
- the equality symbol $=$ naming the well known equality relation;
- the connectives $\neg$ and $\vee$ and the existential quantifier $\exists$.

The non-logical symbols of $\mathcal{L}_n$ are:

- countably many constant symbols, or simply constants: $a, b, \ldots$;

- unary and binary predicate symbols.

$\mathcal{L}_n$ also includes predicate symbols required to represent and reason about time in narratives. We defer the discussion of those symbols, and the axioms that define them, until Section 5.2.3.

The terms of $\mathcal{L}_n$ are constants and variables. The atoms of $\mathcal{L}_n$ are expressions of the form $P(t_1, \ldots, t_k)$, where each t_i is a term. A ground atom is an atom $P(t_1, \ldots, t_k)$ where each t_i is a constant. A formula of $\mathcal{L}_n$ is one of the following:

- an atom;
- a co-reference formula of the form $(t_1 = t_2)$, where t_1 and t_2 are terms;
- the negation of a formula $\neg\alpha$;
- the disjunction of two formulas $(\alpha \vee \beta)$;
- an existential quantification of the form $\exists x.\alpha$

A sentence of $\mathcal{L}_n$ is a formula whose variables, if any, are each bound to one quantifier, i.e. a formula with no free variables. As customary, we will consider as part of $\mathcal{L}_n$ sentences including the universal quantifier $\forall$ and the connectives $\wedge$ ("and") and $\rightarrow$ ("implies"), obtained as abbreviations of the equivalent sentences using the previously introduced symbols. Furthermore, to simplify the notation we omit universal quantifiers in formulae.

All predicate symbols denote pairwise disjoint sets, i.e.:

$$A(x) \rightarrow \neg B(x) \tag{5.1}$$

$$P(x, y) \rightarrow \neg R(x, y) \tag{5.2}$$

where A and B stand for any two different unary predicate symbols, and P and R stand for any two different binary predicate symbols.

The following equality axioms hold in $\mathcal{L}_n$:

$$x = x \tag{5.3}$$

$$(x = y) \rightarrow (y = x) \tag{5.4}$$

$$[(x = y) \wedge (y = z)] \rightarrow (x = z) \tag{5.5}$$

$$(x = y) \rightarrow [A(x) \equiv A(y)] \tag{5.6}$$

$$[(x_1 = y_1) \wedge (x_2 = y_2)] \rightarrow [P(x_1, y_1) \equiv P(x_2, y_2)] \tag{5.7}$$

where A and P are as above.

We adopt the standard first-order semantics to assign meaning to the formulas of the $\mathcal{L}_n$language.

5.2.2 Axioms of the Ontology

In this section, we give the axioms of the NOnt ontology, including those of its subparts NOntNar and NOntDL, expressed using the $\mathcal{L}_n$language described above.

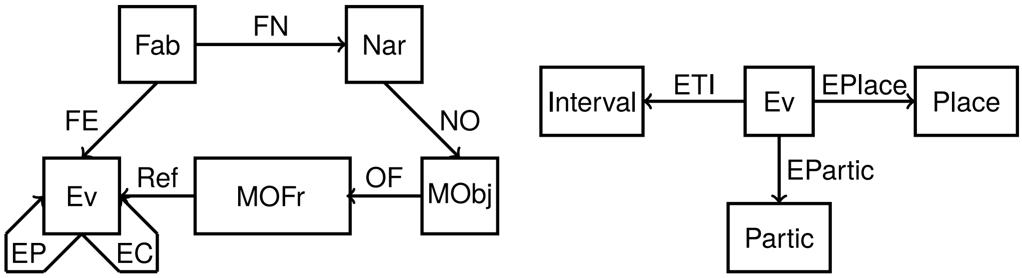

Figure 5.1: *The NOntNar ontology (left) and the predicates related to events (right)*

Axioms of NOntNar The unary and binary predicates of NOntNar are listed in Table 5.1.

Figure 5.1 show a graphical representation of these predicates, including the main predicates of the ontology (left) and a close-up view of the predicates related to events (right).

A concrete example of narrative, selected from the biography of Gustav Klimt, is reported at the end of the present section.

In the following, we list all the axioms holding on the unary and binary predicates of NOntNar.

The following axioms provide the domain and range of each of the binary predicates:

$$\mathsf{EP}(x,y) \rightarrow \mathsf{Ev}(x) \wedge \mathsf{Ev}(y) \quad (5.8)$$

$$\mathsf{EC}(x,y) \rightarrow \mathsf{Ev}(x) \wedge \mathsf{Ev}(y) \quad (5.9)$$

$$\mathsf{ETI}(x,y) \rightarrow \mathsf{Ev}(x) \wedge \mathsf{Interval}(y) \quad (5.10)$$

$$\mathsf{NO}(x,y) \rightarrow \mathsf{Nar}(x) \wedge \mathsf{MObj}(y) \quad (5.11)$$

$$\mathsf{OF}(x,y) \rightarrow \mathsf{MObj}(x) \wedge \mathsf{MOFr}(y) \quad (5.12)$$

$$\mathsf{Ref}(x,y) \rightarrow \mathsf{MOFr}(x) \wedge \mathsf{Ev}(y) \quad (5.13)$$

Table 5.1: *Predicates of NOntNar*

Unary Predicate Symbols	
Ev(e)	e is an event
Interval(t)	t is a time interval
Place(p)	p is a place
Partic(c)	c is an entity that participates in an event (e.g. a person, an object, a concept)
Fab(f)	f is a fabula
Nar(a)	a is a narration
MObj(o)	o is a media object
MOFr(r)	r is a media object fragment
Binary Predicate Symbols	
EP(e_1, e_2)	event e_1 is part of event e_2
EC(e_1, e_2)	event e_1 is caused by event e_2
ETI(e, t)	event e occurs at interval t
EPlace(e, p)	event e occurs in place p
EPartic(e, c)	event e has participant c
FE(f, e)	fabula f has event e
FN(f, a)	fabula f has narration a
NO(n, o)	narration n has content o
OF(o, r)	media object o has fragment r
Ref(r, e)	fragment r is about event e
TINC(t_1, t_2)	interval t_1 includes interval t_2
TIP(t_1, t_2)	interval t_1 starts before t_2

The following cardinality restrictions apply to the NOntNar predicates:

- An event has exactly one time interval:

$$\mathsf{Ev}(x) \rightarrow (\exists y)\mathsf{ETI}(x, y) \tag{5.14}$$
$$\mathsf{ETI}(x, y_1) \wedge \mathsf{ETI}(x, y_2) \rightarrow y_1 = y_2 \tag{5.15}$$

- An event has exactly one place:

$$\mathsf{Ev}(x) \rightarrow (\exists y)\mathsf{EPlace}(x, y) \tag{5.16}$$
$$\mathsf{EPlace}(x, y_1) \wedge \mathsf{EPlace}(x, y_2) \rightarrow y_1 = y_2 \tag{5.17}$$

- An event has one or more participants:

$$\mathsf{Ev}(x) \rightarrow (\exists y)\mathsf{EPartic}(x, y) \tag{5.18}$$

- A fabula has one or more events:

$$\mathsf{Fab}(x) \rightarrow (\exists y)\mathsf{FE}(x, y) \tag{5.19}$$

- A fabula has one or more narrations:

$$\mathsf{Fab}(x) \rightarrow (\exists y)\mathsf{FN}(x, y) \tag{5.20}$$

- A narration has exactly one content:

$$\mathsf{Nar}(x) \rightarrow (\exists y)\mathsf{NO}(x, y) \tag{5.21}$$
$$\mathsf{NO}(x, y_1) \wedge \mathsf{NO}(x, y_2) \rightarrow y_1 = y_2 \tag{5.22}$$

- A fragment belongs to exactly one media object:

$$\mathsf{MOFr}(x) \rightarrow (\exists y)\mathsf{OF}(y, x) \tag{5.23}$$
$$\mathsf{OF}(y_1, x) \wedge \mathsf{OF}(y_2, x) \rightarrow y_1 = y_2 \tag{5.24}$$

We do not admit as consistent the narratives in which event parthood and causal dependency are cyclic, i.e. in which an event is a sub- or super-event of itself or some other event, or in which an event is at the same time a cause and an effect of itself or some other event. Since the relations corresponding to these symbols are transitive, by imposing irreflexivity we obtain acyclicity:

$$\mathsf{EC}(x, y) \rightarrow \neg(x = y) \tag{5.25}$$
$$\mathsf{EC}(x, y) \wedge \mathsf{EC}(y, z) \rightarrow \mathsf{EC}(x, z) \tag{5.26}$$
$$\mathsf{EP}(x, y) \rightarrow \neg(x = y) \tag{5.27}$$
$$\mathsf{EP}(x, y) \wedge \mathsf{EP}(y, z) \rightarrow \mathsf{EP}(x, z) \tag{5.28}$$

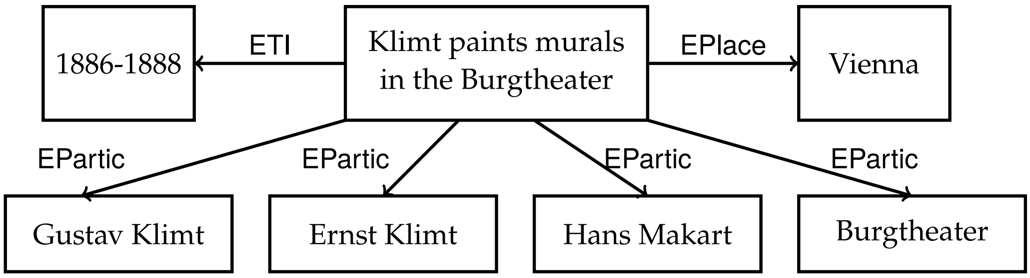

Figure 5.2: *Representation of the event "Klimt paints murals in the Burgtheater"*

The next two axioms rule the interaction of event parthood and causal dependency with time. They state that the period of occurrence of an event is included in the period of occurrence of any of its super-events:

$$\mathsf{EP}(x, y) \wedge \mathsf{ETI}(x, i_x) \wedge \mathsf{ETI}(y, i_y) \rightarrow \mathsf{TINC}(i_y, i_x) \tag{5.29}$$

and that the period of occurrence of an event starts before the period of occurrence of any event that causally depends on it:

$$\mathsf{EC}(x, y) \wedge \mathsf{ETI}(x, i_x) \wedge \mathsf{ETI}(y, i_y) \rightarrow \mathsf{TIP}(i_y, i_x) \tag{5.30}$$

Finally, a fragment that narrates an event x narrates any super-event of x :

$$\mathsf{EP}(x, y) \wedge \mathsf{Ref}(z, x) \rightarrow \mathsf{Ref}(z, y) \tag{5.31}$$

In the following, we report an example of a simple narrative composed of two events, and we show a graph-based representation of this narrative using the axioms defined above. The narrative is based on the following text from Wikipedia: "Between 1886 an 1888 Gustav Klimt, along with his brother Ernst and his friend Hans Makart, painted the murals in the Burgtheater of Vienna. In 1888, Klimt received the Golden Order of Merit from Emperor Franz Josef I of Austria for his contributions to murals painted in the Burgtheater in Vienna"[1].

This text constitutes the *narration* of the narrative, and is expressed in our ontology using the predicate Nar. The *media object* that contains the narration is the section of the Wikipedia page from which the text was extracted, and it is represented using the predicate MObj.

The *fabula* of the narrative is composed of two events, each represented using the predicate Ev. The first event, "Klimt paints murals in the Burgtheater", is represented by the following textual fragment (MOFr): "Between 1886 an 1888 Gustav Klimt, along with his brother Ernst and his friend Hans Makart, painted the murals in the Burgtheater of Vienna".

This event has exactly one time interval (1886-1888), and one place (Vienna). Furthermore, the event has four participants: three people (Gustav Klimt, Ernst Klimt and Hans Makart) and one building (the Burgtheater). A graph-based representation of this event is shown in Figure 5.2.

The second event, "Klimt receives Golden Order of Merit", is represented by the following textual fragment (MOFr): "In 1888, Klimt received the Golden

[1]Text from the English Wikipedia, https://en.wikipedia.org/wiki/Gustav_Klimt

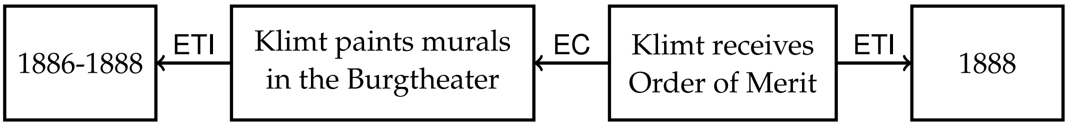

Figure 5.3: *Representation of two events and the causality relation between them*

Order of Merit from Emperor Franz Josef I of Austria for his contributions to murals painted in the Burgtheater in Vienna".

This event has exactly one time interval (the year 1888), and one place (Vienna). The event has three participants: Emperor Franz Josef I, Gustav Klimt and the Golden Order of Merit. The two events are connected through the binary predicate EC, which expresses a causality relation between them. This relation is shown in Figure 5.3.

Figure 5.4: *A view of the NOntDL ontology*

Axioms of NOntDL Table 5.2 lists the unary and binary predicates of the NOntDL ontology. A graphical representation of these predicates is provided in Figure 5.4.

In the following, we list all the axioms holding on the unary and binary predicates of NOntDL.

The two unary predicate symbols are pairwise disjoint:

Table 5.2: *Predicates of NOntDL*

Unary Predicates	
Nrt(n)	n is a narrative
NGraph(g)	g is a narrative graph
Binary Predicates	
NG(n, g)	narrative n has graph g

$$\mathsf{Nrt}(x) \rightarrow \neg\mathsf{NGraph}(x) \tag{5.32}$$

$$\mathsf{NGraph}(x) \rightarrow \neg\mathsf{Nrt}(x) \tag{5.33}$$

Narratives and graphs are one-to-one:

$$\mathsf{NG}(n, g_1) \land \mathsf{NG}(n, g_2) \rightarrow g_1 = g_2 \tag{5.34}$$

$$\mathsf{NGraph}(g) \rightarrow (\exists n)\mathsf{NG}(n, g) \tag{5.35}$$

$$\mathsf{NG}(n_1, g) \land \mathsf{NG}(n_2, g) \rightarrow n_1 = n_2 \tag{5.36}$$

$$\mathsf{Nrt}(n) \rightarrow (\exists g)\mathsf{NG}(n, g) \tag{5.37}$$

A digital library is any KB that includes the above axioms and a set of assertions that connect each narrative to the corresponding NGraph through the NG property. As such, these assertions link the digital library to the graphs containing the formal representations of the narratives that it contains.

5.2.3 Representation of temporal knowledge

NOnt includes predicates for representing qualitative temporal knowledge about the intervals of occurrence of events. Using these predicates, it can be stated, for instance, that an event occurred before, or during, another event. For this purpose, our ontology relies on 13 basic temporal relations (BTRs for short), proposed in the seminal work of Allen (1983) to capture all possible ways in which two intervals can relate to each other.

Allen also provided transitivity rules that allow deriving implicit temporal relations from known ones. These transitivity rules can indeed be expressed in one of the Semantic Web languages, the SWRL rule language. However, the temporal relations between two intervals become exponentially many if disjunctions of BTRs are used to state temporal knowledge; and in the case of narratives, these disjunctions are needed, as will be shown in Section 6.5. Consequently, transitivity rules become exponentially many, making reasoning over narratives practically impossible.

Tractable subsets of BTRs are known to exist for certain domains, but unfortunately, none of these known subsets can be applied to our case, as will be shown in due course. Therefore, we had to search for a tractable subset of BTRs, so as to provide a complete axiomatization of our ontology in SWRL, enabling efficient reasoning on narratives.

As stated in Chapter 4, we represent time in narratives using intervals. Sometimes, the time points giving the beginning and the end of such intervals are known, and the total ordering relation between time points can be used to express and reason over temporal knowledge in a narrative. However, this is not always the case: in many situations, only the relative relation between intervals is known, such that an event occurs before, or during another event. In these cases, a relative form of representation is the only viable option. We therefore need a conceptualization of time that supports both time points and intervals, and absolute and relative relations between them.

Our conceptualization includes both time instants and time intervals, along with the following relations:

- two functions connecting a time interval to its beginning and ending time instants, respectively;
- the total ordering between instants;
- the 13 jointly exhaustive and pairwise disjoint relations in Allen's algebra (Allen, 1983) capturing all possible ways in which two intervals can stand to each other in relative terms. In what follows, we shall call these 13 relations *basic temporal relations* (BTRs, for short). They are as follows: equals (abbreviated as *e*); before (*b*); after (*bi*); meets (*m*); metBy (*mi*); overlaps (*o*); overlappedBy (*oi*); during (*d*); include (*di*); starts (*s*); startedBy (*si*); finishes (*f*); finishedBy (*fi*).

In the previous Section, two more relations between time intervals have been introduced, named in $\mathcal{L}_n$ by the TINC and TIP predicate symbols. These

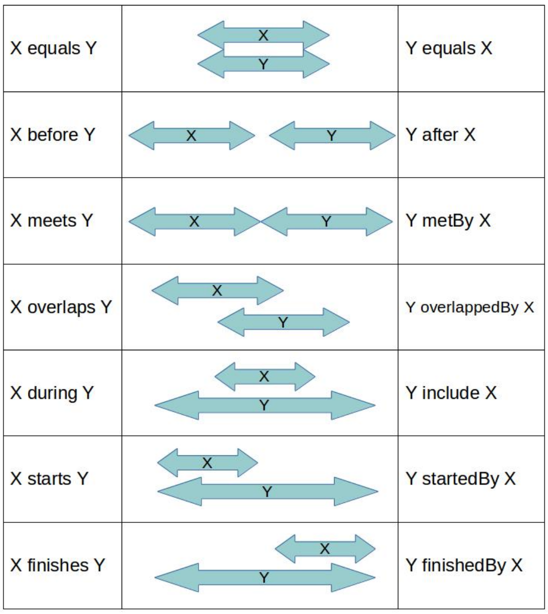

Figure 5.5: *An illustration of Allen's relations between time intervals.*

relations can be expressed as the union of BTRs as follows (for simplicity we abuse notation and use the predicate symbols also for the respective relation):

$$\text{TINC} = \cup \{e, d, s, f\} \quad (5.38)$$

$$\text{TIP} = \cup \{b, m, o, di, fi\} \quad (5.39)$$

Reasoning over Allen's temporal relations has been extensively studied in the literature. For reasons of space, we hereby report only the results of these studies that are relevant to the present context. The interested reader may consult e.g. Renz and Nebel (2007) for a general treatment of this subject, and (Batsakis et al., 2017) for a discussion of temporal reasoning in the context of Semantic Web languages and technologies.

Following Allen's seminal work, the relations between the time intervals in a narrative are maintained in a network, which will be called Qualitative Temporal Knowledge network (QTK for short).

The nodes of a QTK represent the time intervals in the narrative, while the arcs represent relations between the intervals corresponding to the conjoined nodes. The arcs are labelled with non-empty sets of BTRs, and each such set represents the union of its member relations. Specifically, an arc between nodes I and J is labelled by a set L of BTRs if and only if the temporal knowledge stored in the network implies that I and J are related by one of the relations in L. For example, the QTK given in Figure 5.6 stores knowledge about three intervals I, J and K, such that I meets or overlaps both J and K, while J starts K.

At the beginning, a QTK is empty. When a set of relations R between two intervals I and J must be asserted, two nodes corresponding to I and J are created, and the arc between them is added, labelled by R. Now suppose

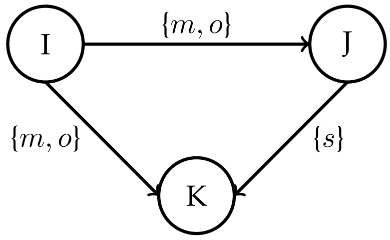

Figure 5.6: *An example of QTK network*

relation set S between nodes J and K needs to be asserted (such as $\{s\}$ in Figure 5.6). Correspondingly, node K is added to the network and S is used as a label of the arc connecting J and K.

Node K must also be connected to all the other nodes of the network by adding the corresponding arcs, each with the appropriate label. In the example, K must be connected to node I with the appropriate label. In absence of any knowledge, the label is the complete set of Allen's relations, meaning that I and K can be related in *any possible way*.

However, the known relations between nodes I and J and between J and K may *restrict* the possible relations between I and K. In order to compute these restrictions, *composition* rules are used. A composition rule is a statement about three intervals I, J and K. The statement has a *premise* and a *conclusion* as follows:

1. the premise gives a set of temporal relations between intervals I and J, and a set of temporal relations between intervals J and K;

2. the conclusion gives a set of temporal relations between intervals I and K.

The meaning of a composition rule is the following: if the relations in the premise hold between nodes I and J and between nodes J and K, then only one of the relations in the conclusion may hold between nodes I and K. Every time a QTK network is updated with additional knowledge, the composition rules are applied in order to restrict the relations labelling the arcs of the network.

In order to see how, let us consider the QTK in Figure 5.6 just after the addition of the knowledge that interval J starts (s) interval K, as shown in Figure 5.7. The arc connecting nodes I and K is labelled with the set of all 13 Allen's relations, indicated by T. To restrict T to only the possible relations that may hold between I and K we use a rule that has as premises the relation sets on arcs I-J and J-K, that is {m, o} and {s} respectively. By reasoning on intervals with the Allen's relations, it is not difficult to see that the conclusion of this rule is in fact the set {m, o}, therefore the resulting QTK network is the one in Figure 5.6.

Given that there are $2^{13} - 1$ non-empty temporal relation sets, and that there is a different composition rule for every pair of such sets, there are millions of composition rules. However, composition rules enjoy a nice mathematical property: the conclusions of the rules having non-singleton premises can be efficiently computed from those of the rules having singleton premises. Since the latter kind of rules are of the order of dozens (they are given in (Allen, 1983)),

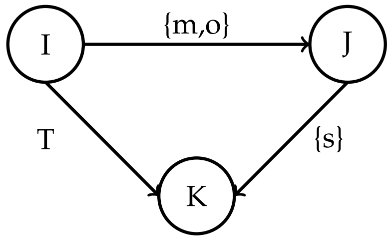

Figure 5.7: *A QTK network including the set of all 13 Allen's relations (T)*

we have a method to efficiently compute the label of any arc of a QTK. However, since the number of possible labels grows exponentially with the number of labelled arcs, and labels need to be re-computed at each update, it may take an exponential amount of time to compute the QTK resulting from an update. This combinatorial explosion is one problem with QTKs.

This is due to the fact that new labels may be generated for some arcs, which in turn cause new labels to propagate to other arcs of the QTK, and so on.

To see the problem, suppose that relation b is asserted between intervals I and K in Figure 5.6. Such relation produces an inconsistent QTK due to the fact that b is incompatible with both the already asserted relations m and o. Likewise, an inconsistency may arise from the application of a composition rule that derives, e.g., relation b for nodes I and K.

A second problem is given by the rise of inconsistencies in QTK. These inconsistencies can be detected by applying *path consistency* (Renz & Nebel, 2007), a technique based on the application of the iterative formula for computing R(I,J)$^{n+1}$, that is the label on the arc between any two nodes I and J at step $n+1$, given the labels between any two nodes at step n. The formula is given by:

$$R(I,J)^{n+1} = R(I,J)^n \cap (\cup_{\mathrm{K}}(R(I,K)^n \circ R(K,J)^n)) \tag{5.40}$$

where $\circ$ denotes the composition of sets of BTRs. The formula is applied iteratively to a QTK until a fixed point is reached, i.e., until the application of the formula does not produce any change in the QTK. If some label equals the empty set, then the QTK is inconsistent. Otherwise, the QTK resulting from path consistency contains a set of labels that is no larger than the set of labels of the initial QTK, and that embodies the temporal knowledge currently held in the network. Path consistency can achieve its task in a polynomial amount of time, therefore the second problem does not prevent the efficient management of a QTK.

In order to address the former problem, it is necessary to seek tractable sets of temporal relations, i.e. sets $\mathcal{T}$ including disjunctions of BTRs such that $\mathcal{T}$ is closed under intersection and composition, so that the application of path consistency always yields a relation in $\mathcal{T}$. This property prevents the combinatorial explosion of the time needed to compute a QTK following an update, while guaranteeing detection of inconsistencies.

In order to perform temporal reasoning over narratives, we have derived a tractable set of temporal relations including the 13 Allen's BTRs and the disjunctions TINC and TIP that we need in order to axiomatize narratives, as explained in Section 5.2.3. This set, which we call $\mathcal{T}_n$, only includes 81 disjunctions; in the remaining part of this section we briefly describe its composition and the way it has been derived.

Minimal tractable set of BTRs We started from the minimal tractable set of BTRs computed in (Batsakis et al., 2017). The set consists of 28 relations, including the 13 primitive ones plus 15 disjunctions:

{*a*}, {*a, d, di, o, oi, mi, s, si, f, fi, eq*}, {*a, d, oi, mi, f*}, {*a, di, oi, mi, si*}, {*a, oi, mi*}, {*b*}, {*b, d, di, o, oi, m, s, si, f, fi, eq*}, {*b, d, o, m, s*}, {*b, di, o, m, fi*}, {*b, o, m*}, {*d*}, {*d, di, o, oi, s, si, f, fi, eq*}, {*d, o, s*}, {*d, oi, f*}, {*di*}, {*di, o, fi*}, {*di, oi, si*}, {*eq*}, {*f*}, {*fi*}, {*f, fi, eq*}, {*m*}, {*mi*}, {*o*}, {*oi*}, {*s*}, {*s, si, eq*}, {*si*}

This set includes the TINC disjunction $\{b, m, o, di, fi\}$, representing the precedence relation, but unfortunately it lacks the TIP disjunction $\{e, d, s, f\}$, representing the inclusion relation. Therefore, this set is not sufficient for our purposes.

In order to solve this issue, we re-computed the minimal tractable set that includes TIP and TINC, using the path consistency algorithm described and implemented in (Batsakis et al., 2017). In particular, given three nodes I, J and K such that I and J stand in relation $r1$ and J and K stand in relation $r2$, the relation between intervals I and K is given by a transitivity table, i.e. a 13 by 13 array whose entry $(r1, r2)$ gives the composition between the two relations.

The path consistency algorithm starts from an initial set of relations, and from the known transitivity table expressing their compositions. Each time a composition results in a new disjunction not present in the set, the algorithm adds a new row to the transitivity table and computes the composition between this disjunction and each other relation. When no new disjunctions are generated, the execution of the algorithm is stopped and the resulting set of relations is returned to the user. In our case, the initial set given as input to the algorithm contains the 13 primitive BTRs, plus TIP and TINC.

At the end of the process, the resulting set contains 81 relations[2]:

{*a*}, {*a, d, di, o, oi, m, mi, s, si, f, fi, eq*}, {*a, d, di, o, oi, mi, s, si, f, fi, eq*}, {*a, d, oi, mi, f*}, {*a, d, oi, mi, s, si, f, eq*}, {*a, di, oi, mi, si*}, {*a, di, oi, mi, si, f, fi, eq*}, {*a, mi*}, {*a, oi, mi*}, {*a, oi, mi, f*}, {*a, oi, mi, si*}, {*a, oi, mi, si, f, eq*}, {*b*}, {*b, d, di, o, oi, m, mi, s, si, f, fi, eq*}, {*b, d, di, o, oi, m, s, si, f, fi, eq*}, {*b, d, o, m, s*}, {*b, d, o, m, s, f, fi, eq*}, {*b, di, o, m, fi*}, {*b, di, o, m, s, si, fi, eq*}, {*b, m*}, {*b, o, m*}, {*b, o, m, fi*}, {*b, o, m, s*}, {*b, o, m, s, fi, eq*}, {*d*}, {*d, di, o, oi, m, mi, s, si, f, fi, eq*}, {*d, di, o, oi, m, s, si, f, fi, eq*}, {*d, di, o, oi, mi, s, si, f, fi, eq*}, {*d, di, o, oi, s, si, f, fi, eq*}, {*d, f*}, {*d, oi, f*}, {*d, oi, mi, f*}, {*d, oi, mi, s, si, f, eq*}, {*d, oi, s, si, f, eq*}, {*d, o, m, s*}, {*d, o, m, s, f, fi, eq*}, {*d, o, s*}, {*d, o, s, f, fi, eq*}, {*d, s*}, {*d, s, f, eq*}, {*di*}, {*di, fi*}, {*di, oi, mi, si*}, {*di, oi, mi, si, f, fi, eq*}, {*di, oi, si*}, {*di, oi, si, f, fi, eq*}, {*di, o, fi*}, {*di, o, m, fi*}, {*di, o, m, s, si, fi, eq*}, {*di, o, s, si, fi, eq*}, {*di, si*}, {*di, si, fi, eq*}, {*eq*}, {*f*}, {*f, eq*}, {*f, fi, eq*}, {*fi*}, {*fi, eq*}, {*m*}, {*mi*}, {*o*}, {*o, fi*}, {*o, m*}, {*o, m, fi*}, {*o, m, s*}, {*o, m, s, fi, eq*}, {*o, s*}, {*o, s, fi, eq*}, {*oi*}, {*oi, f*}, {*oi, mi*}, {*oi, mi, f*}, {*oi, mi, si*}, {*oi, mi, si, f, eq*}, {*oi, si*}, {*oi, si, f, eq*}, {*s*}, {*s, eq*}, {*s, si, eq*}, {*si*}, {*si, eq*}.

[2]These relations are defined at the beginning of the SWRL rule file at the following address: https://dlnarratives.eu/ontology/swrl-rules.owl

In order to reason on these 81 relations, it is necessary to explicitly express as rules all the possible compositions and intersections between each pair of relations contained in the set. In theory, this process should yield 6,561 composition rules plus 6,561 intersection rules, for a total of 13,122 rules.

In practice, however, many rules can be safely removed because they involve $\{a, b, d, di, o, oi, m, mi, s, si, f, fi, eq\}$, i.e. the disjunction of all basic relations. This disjunction always holds between two intervals, thus it does not add any new information to the graph. By removing the rules involving this disjunction, the final number of rules is reduced to 7,671.

5.2.4 Axioms for temporal representation

We can now complete the expression of NOnt by introducing and axiomatising the symbols for temporal representation and reasoning.

Table 5.3 gives the unary temporal predicate symbols, while Table 5.4 gives the binary ones. As shown in Table 5.3, NOnt provides time points in addition to time intervals, for usability in realistic contexts. Consequently, the symbols modeling the ordering of time points and those for linking intervals to their beginning and ending time points are added as well.

Table 5.3: *Unary temporal predicates*

Ev(e)	e is an event
Interval(t)	t is a time interval
Fab(f)	f is a fabula
Nar(a)	a is a narration
MObj(o)	o is a media object
MOFr(r)	r is a media object fragment

The binary predicate T stands for each of the 81 binary predicate symbols that are one-to-one with the relations in $\mathcal{T}_n$, allowing users to exploit the full power of the temporal language in manipulating narratives. Finally, the special predicate symbol $\perp$ stands for the empty relation.

The following axioms provide domain, range and cardinality of the symbols linking intervals and time points:

Table 5.4: *Binary temporal predicates*

IB(t, p)	interval t begins at point p
IE(t, p)	interval t ends at point p
$p_1 < p_2$	point p_1 precedes point t_2
$p_1 = p_2$	point p_1 is equal to point t_2
$p_1 > p_2$	point p_1 follows point t_2
T(t_1, t_2)	interval t_1 is in relation T with interval t_2
$\perp$	the empty relation symbol

$$\mathsf{Interval}(x) \rightarrow (\exists b)\mathsf{IB}(x, b) \tag{5.41}$$

$$\mathsf{IB}(x, b) \rightarrow \mathsf{Interval}(x) \wedge \mathsf{TPoint}(b) \tag{5.42}$$

$$\mathsf{IB}(x, b_1) \wedge \mathsf{IB}(x, b_2) \rightarrow b_1 = b_2 \tag{5.43}$$

$$\mathsf{Interval}(x) \rightarrow (\exists e)\mathsf{IE}(x, e) \tag{5.44}$$

$$\mathsf{IE}(x, e) \rightarrow \mathsf{Interval}(x) \wedge \mathsf{TPoint}(e) \tag{5.45}$$

$$\mathsf{IE}(x, e_1) \wedge \mathsf{IE}(x, e_2) \rightarrow e_1 = e_2 \tag{5.46}$$

The axioms on the symbols for ordering time points are not given, since the corresponding relations are constants and are available in any implementation.

The axioms on the symbols standing for the relations in $\mathcal{T}_n$ are given by the following sets of formulas:

- the set $\mathcal{C}_t$ containing the sentences for the composition of the temporal predicate symbols, each having the form

$$\mathsf{R}_1(x,y) \wedge \mathsf{R}_2(y,z) \rightarrow \mathsf{R}_3(x,z) \tag{5.47}$$

where each R_i is a temporal predicate symbol;

- the set $\mathcal{I}_t$ containing the sentences for the intersection of the temporal predicate symbols, each having the form

$$\mathsf{R}_1(x,y) \wedge \mathsf{R}_2(x,y) \rightarrow \mathsf{R}_3(x,y) \tag{5.48}$$

where each R_i is a temporal predicate symbol, and R_3 can be $\bot$;

- the set $\mathcal{P}_t$ containing the sentences relating the symbols of the 13 BTRs on time intervals and those on time points. Each such sentence is an if-and-only-if statement, expressing equivalence of one of the Allen's BTRs with a conjunction of atoms on the symbols $<$, $=$, and $>$ on time points. All these share the sub-formula

$$\mathsf{IB}(x,b_x) \wedge \mathsf{IE}(x,e_x) \wedge \mathsf{IB}(y,b_y) \wedge \mathsf{IE}(y,e_y) \tag{5.49}$$

which binds the six involved variables in the appropriate way. By abbreviating this formula as $\alpha(x,y)$, the sentence for the before BTR is given by:

$$\alpha(x,y) \rightarrow (\mathsf{before}(x,y) \equiv e_x < b_y) \tag{5.50}$$

which is equivalent to the two implications

$$(\alpha(x,y) \wedge \mathsf{before}(x,y)) \rightarrow e_x < b_y \tag{5.51}$$

$$(\alpha(x,y) \wedge e_x < b_y) \rightarrow \mathsf{before}(x,y) \tag{5.52}$$

As we have seen in Section 6.4, the binary predicate TIP representing a causality relation between two events implies that the corresponding temporal intervals are linked by one of the relations from the set $\{b, m, o, di, fi\}$. For example, suppose that for the two events reported in Figure 4 we do not know the exact beginning and end of the temporal intervals, but we know that the first event causes the second one. From this knowledge, we can infer that the BTR existing between the two time intervals must be one of the relations from the aforementioned set.

Analogously, the two axioms for the overlaps BTR are given by (omitting $\alpha(x,y)$ for simplicity):

$$\mathsf{overlaps}(x,y) \rightarrow (b_x < b_y \wedge e_x < e_y \wedge e_x > b_y) \tag{5.53}$$

$$(b_x < b_y \wedge e_x < e_y \wedge e_x > b_y) \rightarrow \mathsf{overlaps}(x,y) \tag{5.54}$$

This concludes our definition of the axioms of NOnt. In the following section, we will discuss the implementation of the ontology through the languages of the Semantic Web.

5.3 Implementing NOnt Using Semantic Web Technologies

As we discussed in Section 2.3, ontologies have long been recognized to be a crucial component of the Semantic Web (Antoniou & van Harmelen, 2004). The recommendation of languages for expressing ontologies is a core activity of the W3C, which has produced a whole family of powerful such languages, collectively known as Ontology Web Language (OWL for short) (W3C OWL Working Group, 2012), directly derived from Description Logics.

The OWL family has now reached its second generation, OWL 2. It is therefore natural to consider the most expressive decidable language of the OWL family, OWL 2 DL, as a candidate for implementing NOnt.

In this respect, we would implement unary predicate symbols as OWL 2 DL classes, and binary predicate symbols as OWL 2 DL object or data properties, depending on whether the range of a property is a class or a datatype. A wide array of datatypes are also available in OWL 2 DL, including the XML Schema datatype `dateTime`, which would be a most natural candidate for the implementation of time points.

Based on this correspondence, the axioms of NOnt would have to be translated into OWL 2 DL axioms, by relying on the rich variety of operators that OWL 2 DL offers to this end. Before considering such translation, however, there are two immediate reasons why OWL 2 DL is not sufficient for this task:

1. Properties corresponding to the EC and EP predicate symbols would have to be declared as irreflexive and transitive, to correctly reflect axioms 5.25 to 5.29 of NOnt. However, transitive properties are composite in an OWL 2 DL ontology, and as such they cannot be declared to be irreflexive, so as not to violate the global restrictions on the axioms of an OWL 2 DL ontology (Motik et al., 2012).

2. Path consistency requires axioms for the composition of temporal properties (given in set $\mathcal{C}_t$). These axioms can be expressed in OWL 2 DL as complex role inclusions. Now, the properties that occur in the right-hand side of complex role inclusions are composite, and this would prevent the expression of important axioms on these properties, for instance the axioms stating disjointness from other properties.

Furthermore, declaring the composition of the 81 temporal properties in $\mathcal{T}_n$ would require thousands of complex role inclusion axioms, and it would most certainly be impossible to avoid circular definitions, as required by the global restrictions on the axioms of an OWL 2 DL ontology[3].

Loosely speaking, two complex role inclusion axioms form a circular definition if one of them has property P in the head and property Q in the body, while the other has property Q in the head and property P, or a property used to define P, in the body.

[3] https://www.w3.org/TR/owl2-syntax/#Global_Restrictions_on_Axioms_in_OWL_2_DL

An alternative to OWL 2 DL, which has also been considered in (Batsakis et al., 2017), is the Semantic Web Rule Language (SWRL)[4], a language of the Semantic Web family for specifying Horn clauses (Lloyd, 1984).

We recall that a Horn clause is a definite program clause (DPC) or a definite goal. A DPC r is a $\mathcal{L}_n$ sentence of the form

$$r:\ B_1 \wedge \ldots \wedge B_n \rightarrow A,\ \ n \geq 0 \tag{5.55}$$

where each B_i and A are atoms. The conjunction $B_1 \wedge \ldots \wedge B_n$ is the *body* of the DPC r, while A is the *head*. A DPC clearly resembles a rule, whence the name of the language. If $n = 0$, r is given by

$$r : \rightarrow A \tag{5.56}$$

and is said to be a *unit* clause; a unit clause is just a notational variant for the atom A. Finally, a definite goal is a DPC with no head.

In order to implement NOnt using SWRL, the axioms of the ontology must be expressed as definite program clauses (DPCs). In fact, most of these axioms are *already* DPCs (such as for instance the axioms in $\mathcal{C}_t$, in $\mathcal{I}_t$, or in $\mathcal{P}_t$). Some of the remaining axioms can be easily transformed into DPCs. This is the case of axioms that are implications with a conjunction in their consequent, such as axioms 5.8 to 5.13. Each such axiom is equivalent to DPCs that have as body the antecedent of the implication, and as head a different conjunct in the consequent of the implication. For instance, axiom

$$\mathsf{EP}(x, y) \rightarrow \mathsf{Ev}(x) \wedge \mathsf{Ev}(y) \tag{5.57}$$

is equivalent to the DPCs

$$\mathsf{EP}(x, y) \rightarrow \mathsf{Ev}(x) \tag{5.58}$$

$$\mathsf{EP}(x, y) \rightarrow \mathsf{Ev}(y) \tag{5.59}$$

In addition, axioms that have an equivalence in the head can also be easily transformed into DPCs. These axioms are of the form

$$B_1 \wedge \ldots \wedge B_n \rightarrow (A \equiv A') \tag{5.60}$$

like axioms 5.6 and 5.7. Each such axiom is equivalent to the pair of DPCs

$$B_1 \wedge \ldots \wedge B_n \wedge A \rightarrow A' \tag{5.61}$$

$$B_1 \wedge \ldots \wedge B_n \wedge A' \rightarrow A \tag{5.62}$$

as we have already argued concerning the axioms in $\mathcal{P}_t$. Finally, the reflexivity axiom for equality can be replaced by the DCP

$$\neg(x = x) \rightarrow \perp \tag{5.63}$$

which produces a contradiction whenever an irreflexive axiom is violated.

[4]https://www.w3.org/Submission/SWRL/

However, axioms containing negation (such as axiom 5.1) or the existential quantifier (such as axiom 5.14) are not trivially reduced to DPCs. The remaining part of this section shows that these axioms can be dealt with using SWRL, which is chosen as the implementation language of the temporal representation aspects of NOnt.

Time points will be implemented as values of the `dateTime` datatype of XML Schema[5], thereby equating the unary predicate symbol TPoint with that datatype.

Eliminating negation Since it does not appear in the body of any rule, negation can be handled without resorting to the techniques devised in Datalog, such as stratification (Abitebul et al., 1995). A much simpler approach is indeed possible (Meghini & Doerr, 2018), which consists in introducing a new set of predicate symbols, called *complements,* that are one-to-one with the predicate symbols in $\mathcal{L}_n$ and that stand for the negation of the corresponding predicate symbols.

Technically, for every predicate symbol P in $\mathcal{L}_n$, we introduce a new predicate symbol called the complement of P. As customary, the complement of the equality symbol $=$ will be denoted as $\neq$, while the complement of any other predicate symbol P will be denoted as $\overline{P}$. We then modify the set of NOnt axioms as follows:

1. replace any instance of the axiom schema 5.1

$$A(x) \rightarrow \neg B(x) \tag{5.64}$$

   by the corresponding instance of the schema:

$$A(x) \rightarrow \overline{B}(x) \tag{5.65}$$

   and add

$$A(x) \wedge \overline{A}(x) \rightarrow \bot \tag{5.66}$$

2. replace any instance of the axiom schema 5.2

$$P(x, y) \rightarrow \neg R(x, y) \tag{5.67}$$

   by the corresponding instance of the schema:

$$P(x, y) \rightarrow \overline{R}(x, y) \tag{5.68}$$

   and add

$$P(x, y) \wedge \overline{P}(x, y) \rightarrow \bot \tag{5.69}$$

By so doing, a new set of axioms is obtained, which is intuitively equivalent to the initial set, since the two sets state the same sentences in different ways.

[5]https://www.w3.org/TR/xmlschema-2/

Dealing with existential quantification The typical technique for eliminating existentially quantified variables from first-order formulae is Skolemization. Skolemization is performed by replacing every existentially quantified variable y in the scope of n universally quantified variables $x_1, \ldots, x_n$ with a term $f(x_1, \ldots, x_n)$ where f is a new function symbol.

However, Skolemization cannot be applied to reduce a set of axioms to SWRL rules, because function symbols are not allowed in SWRL rules. As a consequence, the existentially quantified axioms of NOnt , which are:

$$\mathsf{Ev}(x) \rightarrow (\exists y)\mathsf{ETI}(x, y) \tag{5.70}$$

$$\mathsf{Nar}(x) \rightarrow (\exists y)\mathsf{NO}(x, y) \tag{5.71}$$

$$\mathsf{MOFr}(x) \rightarrow (\exists y)\mathsf{OF}(x, y) \tag{5.72}$$

$$\mathsf{Interval}(x) \rightarrow (\exists y)\mathsf{IB}(x, y) \tag{5.73}$$

$$\mathsf{Interval}(x) \rightarrow (\exists y)\mathsf{IE}(x, y) \tag{5.74}$$

cannot be transformed into SWRL rules and must therefore be expunged from the SWRL implementation of NOnt.

The negative effect of this elimination can be mitigated by considering that the individuals denoted by the existential variables in the above axioms are all unique, as guaranteed by the corresponding cardinality axioms:

$$\mathsf{ETI}(x, y_1) \land \mathsf{ETI}(x, y_2) \rightarrow y_1 = y_2 \tag{5.75}$$

$$\mathsf{NO}(x, y_1) \land \mathsf{NO}(x, y_2) \rightarrow y_1 = y_2 \tag{5.76}$$

$$\mathsf{OF}(y_1, x) \land \mathsf{OF}(y_2, x) \rightarrow y_1 = y_2 \tag{5.77}$$

$$\mathsf{IB}(x, y_1) \land \mathsf{IB}(x, y_2) \rightarrow y_1 = y_2 \tag{5.78}$$

$$\mathsf{IE}(x, y_1) \land \mathsf{IB}(x, y_2) \rightarrow y_1 = y_2 \tag{5.79}$$

Moreover, the individuals implied by the first three axioms, i.e. the time interval of an event, the content of a narration, and the media object containing a fragment, are all known at the time when the corresponding ETI, NO, and OF atoms are asserted, therefore we will design the interface of the system in a way that forces the user to specify those individuals.

The situation is different for the last two axioms: the starting and ending points of a temporal interval may not be known at the time when the interval is asserted, and this is in fact the reason why NOnt allows the representation and reasoning about qualitative temporal knowledge.

In the last two cases, then, failing the user to provide a data value for each of these points, the system will force temporal constants for them, using these constants as placeholders for the corresponding values about which knowledge can be expressed or inferred by the system.

Table 5.5: *Mapping of NOnt classes to reference ontologies*

Predicate	Class Name	Linked Class
Nrt	Narrative	subclass of E73 Information Object
Fab	Fabula	subclass of E4 Period
Nar	Narration	subclass of F14 Individual Work
Ev	Event	equivalent to E5 Event
MObj	MObject	subclass of F22 Self-Contained Expression
MOFr	MOFragment	subclass of E90 Symbolic Object
Interval	TimeInterval	equivalent to E52 Time-Span
		equivalent to Proper Interval of OWL Time
TPoint	TimePoint	equivalent to Instant of OWL Time

5.4 Mapping of NOnt to Reference Ontologies

The first requirement we took into account to develop our ontology was its semantic interoperability. Semantic interoperability is a two-way concept: on the one hand, we aim at widening the usage of our ontology for narratives, by making it re-usable; on the other, we aim at re-using as much as possible of existing ontologies in developing our own.

Tables 5.5 and 5.6 report the mapping between NOnt and its reference ontologies, for classes and properties respectively. In the tables, the classes starting with E and the properties starting with P are from the CIDOC CRM (Doerr, 2003), while the classes starting with F and the properties starting with R are from FRBRoo (Doerr et al., 2008).

We adopted the CIDOC CRM ontology as reference because it is an ISO standard largely employed in the Digital Libraries and Cultural Heritage domains. The CRM includes temporal entities for capturing time-dependent concepts such as events. The CRM is also integrated with FRBRoo, an ontology that provides fundamental notions for the modeling of bibliographic and textual knowledge.

To represent the temporal dimension of NOnt, we also integrated it with OWL Time (Hobbs & Pan, 2020), a domain ontology recommended by the W3C for the representation of time.

To create the mapping, we analysed the definitions of the classes and properties of the three reference ontologies. In particular, we took into account the following versions of the ontologies: CIDOC CRM 6.2.9[6], FRBRoo 3.0[7], OWL Time W3C Recommendation of 19 October 2017[8].

As can be seen in Table 5.5, the class Narrative is a subclass of E73 Information Object from the CIDOC CRM, while the fabula is considered a subclass of E4

[6]http://www.cidoc-crm.org/Version/version-6.2.9
[7]http://www.cidoc-crm.org/frbroo/ModelVersion/frbroo-v.-3.0
[8]https://www.w3.org/TR/2017/REC-owl-time-20171019/

Table 5.6: *Mapping of NOnt properties to reference ontologies*

Predicate	Property Name	Linked Property
FN	hasFabula	subproperty of P148 has component
FE	hasEvent	subproperty of P9 consists of
NO	hasMediaObject	subproperty of R9 is realised in
OF	hasFragment	subproperty of R15 has fragment
Ref	refersTo	subproperty of P129 is about
EP	hasSubevent	subproperty of P9 consists of
EC	isCausedBy	superproperty of P15 was influenced by
ETI	hasTimeSpan	equivalent to P4 has time-span
EPartic	hasParticipant	equivalent to P12 occurred in the presence of
EPlace	hasPlace	equivalent to P7 took place at
	hasContent	equivalent to P190 has symbolic content

Period (itself a superclass of E5 Event). The class Event is equivalent to E5 Event from the CIDOC CRM.

The class Narration, representing the textual level of the narrative, is represented as a subclass of F14 Individual Work from FRBRoo. The class Narration is represented in its concrete form through a Media Object (subclass of F22 Self-Contained Expression), which is divided into a set of Media Object Fragments (subclass of E90 Symbolic Object[9]).

Finally, the two temporal classes, Time Interval and Time Point, are linked to the corresponding ones in OWL Time (time:ProperInterval and time:Instant, respectively).

As can be seen in Table 5.6, the properties that apply to the fabula are linked to the CRM: hasFabula, which connects connects a narrative to its fabula, is a subproperty of P148 has component; hasEvent, which connects the fabula to its events, is a subproperty of P9 consists of; connects the fabula to its events.

Furthermore, the three properties that connect an event to another event are also linked to the CRM: (i) the mereology property hasSubevent is a subproperty of P9 consists of; (ii) the causal dependency property isCausedBy is a superproperty of P15 was influenced by; (iii) the properties connecting time intervals are linked to the corresponding ones of the CRM (see Section 5.4.1 below).

5.4.1 Additional Classes and Properties for Modeling Time

We provide some additional properties for associating actual data values, i.e. dates and time instants, to the time spans of our events. To this end, we adopt from the OWL Time ontology (Hobbs & Pan, 2020) the following properties, all having time intervals as domain:

[9]We did not use E73 Expression Fragment because it has been deprecated in the newest iteration of FRBRoo (now called LRMoo): http://www.cidoc-crm.org/ModelVersion/lrmoo-f.k.a.-frbroo-v.0.6.

- owl:inXSDDate, ranging on xsd:date
- owl:inXSDDateTimeStamp, ranging on xsd:dateTimeStamp
- owl:inXSDgYear, ranging on xsd:gYear
- owl:inXSDgYearMonth, ranging on xsd:gYearMonth

The datatypes where these properties range also provide the ordering relations between time points (<, =, >).

In addition, we need names for the set of Allen's relations between time intervals (Allen, 1983) (see Figure 5.5). We reuse the names for these properties from the CRM, where they are directly associated with events. These are:

- crm:P114 is equal in time to (X equals Y and Y equals X in Figure 5.5);
- crm:P115 finishes (X finishes Y);
- crm:P115i is finished by (Y finishes X);
- crm:P116 starts (X starts Y);
- crm:P116i is started by (Y starts X);
- crm:P117 occurs during (X during Y);
- crm:P117i includes (Y include X);
- crm:P118 overlaps in time with (X overlaps Y);
- crm:P118i is overlapped in time by (Y overlappedBy X);
- crm:P119 meets in time with (X meets Y);
- crm:P119i is met in time by (Y metBy X);
- crm:P120 occurs before (X before Y);
- crm:P120i occurs after (Y after X).

5.4.2 Additional Classes and Properties for Modeling Space

To provide a more detailed representation of space, we define an additional class for representing spatial regions, called SpatialRegion. This class comprises spatial regions having one of the following forms:

- a point on the surface of the Earth identified by latitude and longitude;
- a polygon as represented by GIS systems;
- a rectangular region on the surface of the Earth identified by its four vertices;
- any region identified by a IRI in a standard gazetteer, such as Geonames for modern places, Pleiades for ancient places.

Each of these representations is assigned to a specific sub-class of SpatialRegion. These subclasses are (see Figure 3 below):

- SpatialRegionPoint, having as instances regions that are points. The properties defined on points are:
- hasLatitude, having xsd:decimal as range, giving the latitude value of the coordinates of the region
- hasLongitude, having xsd:decimal as range, giving the longitude value of the coordinates of the region
- SpatialRegionPolygon, having as instances regions that are polygons, represented in some format typically managed by a GIS. No commitment is made on a specific notation for these regions, they are treated simply as abstract objects with a specific property:
- hasPolygonalRepresentation, having rdf:XMLLiteral as range, giving the XML document representing the polygonal region
- SpatialRegionBBox, having as instances regions that are bounded boxes, each represented by the minimum and maximum latitude and longitude identifying the box. The properties giving these values are:
- hasBoundingBoxMinLat
- hasBoundingBoxMinLon
- hasBoundingBoxMaxLat
- hasBoundingBoxMaxLon

  All these properties range over xsd:decimal.
- SpatialRegionStdName having as instances regions that are simply identified by a standard name in some vocabulary. The standard name is given by the property:
- hasPlaceIRI, having owl:Thing as range

Two properties are defined for the instances of SpatialRegions:

- hasCoordinateSystem, having xsd:string as range, giving a name that identifies the coordinate system used for representing the region
- hasPlaceName, having xsd:string as range, giving a name that identifies the region in some relevant linguistic context

5.4.3 Modeling Actors and Their Roles in Events

When representing narratives, it is useful to specify a role for the actors in an event (e.g. Pliny the Elder is an observer of the Vesuvius eruption). In natural language, the relation between actor, role and event is a triadic relation, since it involves three individuals.

On the other hand, in NOnt we cannot model triadic relations, since ternary properties are not allowed in Semantic Web languages. As we will see, the CRM solves this problem by allowing properties of properties (e.g. P14.1). But again, this solution cannot be adopted in NOnt.

For these reasons, NOnt provides the class ActorWithRole and three properties to properly connect its instances to events, actors and roles. The three properties are (see Figure 5.8):

- property hasParticipant links events to instances of ActorWithRole;
- property hasRole links instances of ActorWithRole to a type giving the role of the actor; and
- property hasActor links instances of ActorWithRole to instances of class Actor, giving the person or the group that participates in the event.

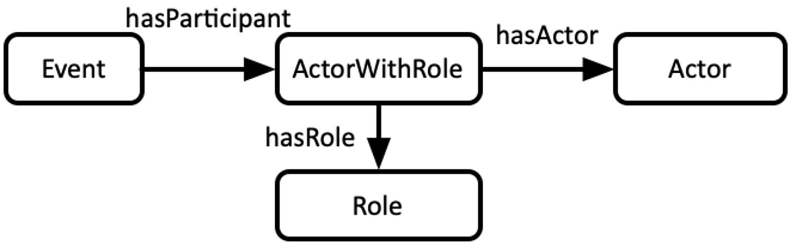

Figure 5.8: *Representation of actor roles in NOnt*

We also introduce the class Person in NOnt, to represent actors in events, or narrators of narratives, that are persons. Person is a subclass of Actor.

Examples of Representation Using the Ontology

Figure 5.9 shows an example of representation of an event in the fabula and narration levels. The event e1 Death of Dante, which happened in Ravenna and had Dante as participant, is connected to a media object fragment containing the text "Dante died in Ravenna". The property refersTo expresses the reference relation existing between the media object fragment ef1.1.1 and the event e1.

Figure 5.10 shows an example of representation of two events in the fabula. The event e2 (Exile of Dante), which happened in Florence, occurs before the event e1 (Death of Dante), which happened in Ravenna. Both events have Dante Alighieri as participant.

Figure 5.11 shows an example of representation of the textual structure of the narration. The narration is expressed through a media object, connected

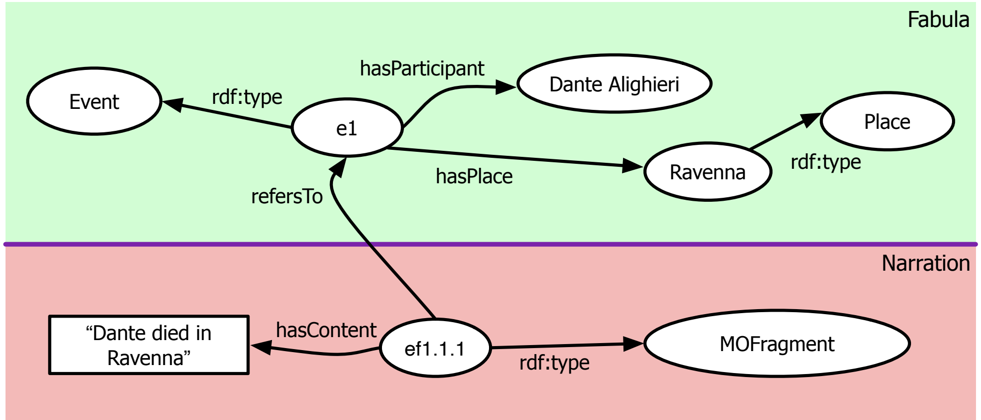

Figure 5.9: *Representation of an event in the fabula and in the narration*

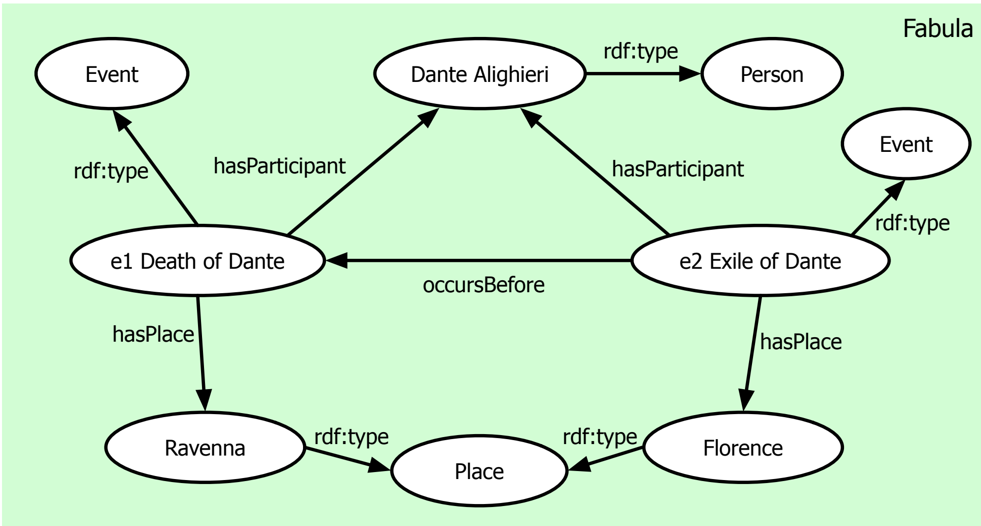

Figure 5.10: *Representation of two events in the fabula*

through the property hasMediaObject, and the media object contains several fragments (in the example, these are fragments of text). Each fragment has several sub-fragments, and each has a content (in this case, a string of text).

The current implementation of the ontology is available on our website, along with the set of SWRL rules that we have implemented for temporal reasoning[10].

[10]https://dlnarratives.eu/ontology/

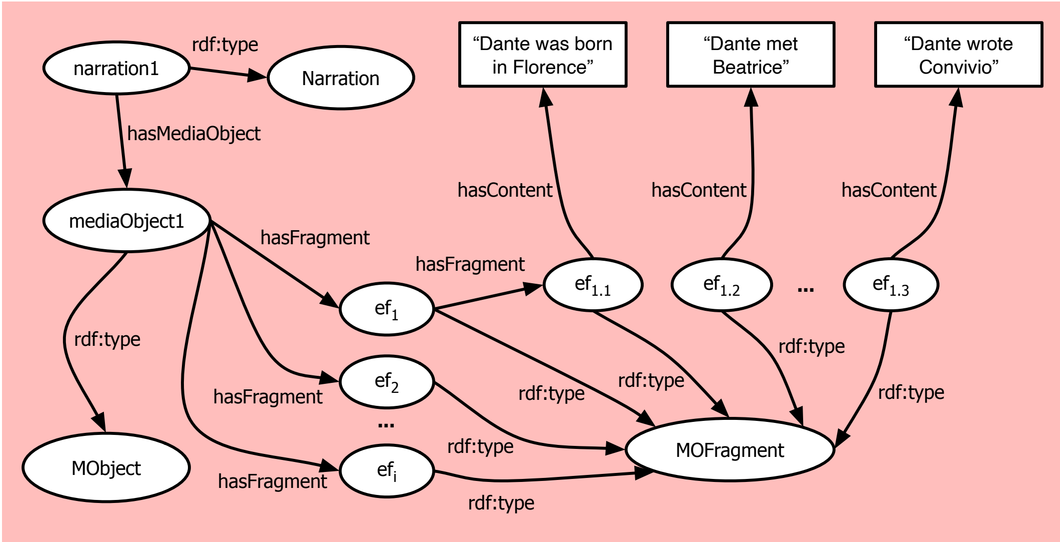

Figure 5.11: *Representation of the textual structure of the narration*

5.5 Extensions to the Ontology

In this section we propose three extensions to NOnt that would allow us to better support the representation of the textual and plot levels of the narratives. The NarraNext tool, presented in Chapter 6, makes use of these extensions.

5.5.1 Integration of NOnt with Linguistic Ontologies

In this section, we present an extension of the Narrative Ontology that allows the representation of the textual level of the narrative by integrating the ontology with standards from the field of Linguistic Linked Open Data.

This is proposed as an extension, instead of a core feature of NOnt, because our ontology aims to be general enough to represent not only narrations in natural language, but also other types of media objects (e.g. video, audio) whose representation is not supported by linguistic ontologies.

As we discussed in the previous section, NOnt is already integrated with FRBRoo, and this integration provides the means to represent some aspects of the textual structure, as shown in Figure 5.11.

Integration with the NIF and GAF ontologies

After analyzing the ontologies reported in Section 2.3.2, we decided to adopt the NIF (Natural language Interchange Format) standard, for the following reasons:

1. NIF is a de facto standard in this field, being used as the basis for numerous other ontologies, e.g. Lemon (J. McCrae et al., 2011), Ontolex (J. P. McCrae et al., 2017), NAF (Fokkens et al., 2014);

2. NIF allows the representation of the concrete textual level of the narrative, including each fragment of text and their position in the overall media object, in a more detailed way than is possible with FRBRoo;
3. unlike some of the other ontologies in this field, NIF is a very general ontology that allows the representation of any kind of text;
4. based on this NIF integration, it will be possible to integrate other ontologies depending on what other aspects of text we might want to represent;
5. given the significant number of ontologies that offer compatibility with NIF, this mapping allows the connection of our narrative to the larger Linguistic Linked Open Data cloud.

.

The integration with NIF allows us to represent several features of the text that cannot be modeled using FRBRoo alone. First of all, NIF allows the representation of sentence structure. Each sentence is made of words, although NIF does not prescribe what a word *is*.

In NIF, text is represented using the class nif:String. Given that our class narra:Fragment is more general (e.g. it could represent a fragment of video), we declare narra:Fragment as a superproperty of nif:String.

Figure 5.12 shows the fabula and the narration integrated with NIF. At the narration level, the property nif:anchorOf allows us connect a fragment of text to the corresponding string (the same we did with our property narra:hasContent, subproperty of P190 has symbolic content). Given that narra:hasContent is applicable to different kinds of media objects, and not just texts, we declare it as a superproperty of the more specific nif:anchorOf.

It should be noted that our reference relation narra:refersTo does not have a NIF equivalent. Due to this fact, in the Newsreader project (Vossen et al., 2014), a new ontology was created specifically to represent this kind of relation between a fragment of text and the resource it denotes. This property is called gaf:denotes, and it is the only property in the GAF ontology (Fokkens et al., 2014), developed in the context of the NewsReader project (Vossen et al., 2014). To further our integration with linguistic ontologies, we have decided to declare our property refersTo equivalent to gaf:denotes.

Table 5.7 reports the mapping of the NOnt classes and properties with the corresponding ones of NIF and GAF.

Table 5.7: *Mapping of NOnt classes and properties with NIF*

Predicate	Class or Property	Linked Class or Property
MOFr	Fragment	superclass of nif:String
OF	hasFragment	superclass of nif:subString
	hasContent	superclass of nif:anchorOf
Ref	refersTo	equivalent to gaf:denotes

In addition, NIF provides two properties, nif:beginIndex and nif:endIndex, that connect a string of text to the index of that string in a longer string that contains

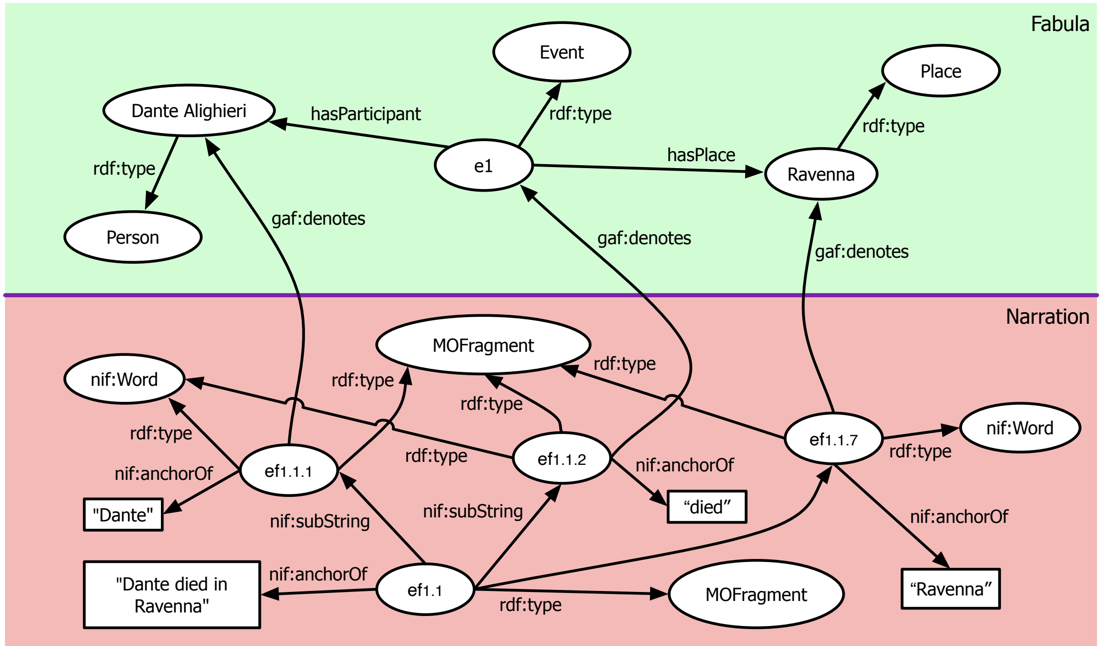

Figure 5.12: *Representation of the fabula and narration using NIF*

it. These properties are very useful to identify the location of a fragment in a parent fragment. Figure 5.13 shows the structure of the narration integrated with NIF.

Representation of Annotation Confidence

Furthermore, NIF provides the nif:classAnnotation property to annotate the fact that a string of text belongs to a particular class. The property nif:confidence allows the representation of a confidence level (between 0 and 1) for the annotation.

This is particularly useful to us when automating narrative extraction tasks, because by applying these properties we can store in our knowledge base the full results of our extraction, without making an *a priori* decision about the validity of each individual classification.

For example, a component of our NarraNext tool may classify the string "Victoria" as a person with confidence 0.6, but this string may also represent a place (the Australian state of Victoria). Instead of deciding immediately whether the interpretation of the tool is correct, we can use NIF to store the annotation and let the user decide if this classification should be added to the KB.

5.5.2 Expressing the Plot

At present, the *plot* level of the narrative is not represented explicitly in the ontology, because the reference function allows us to derive it from the fabula and the narration(s). Indeed, the order of events in one narration expresses the plot of that specific narration.

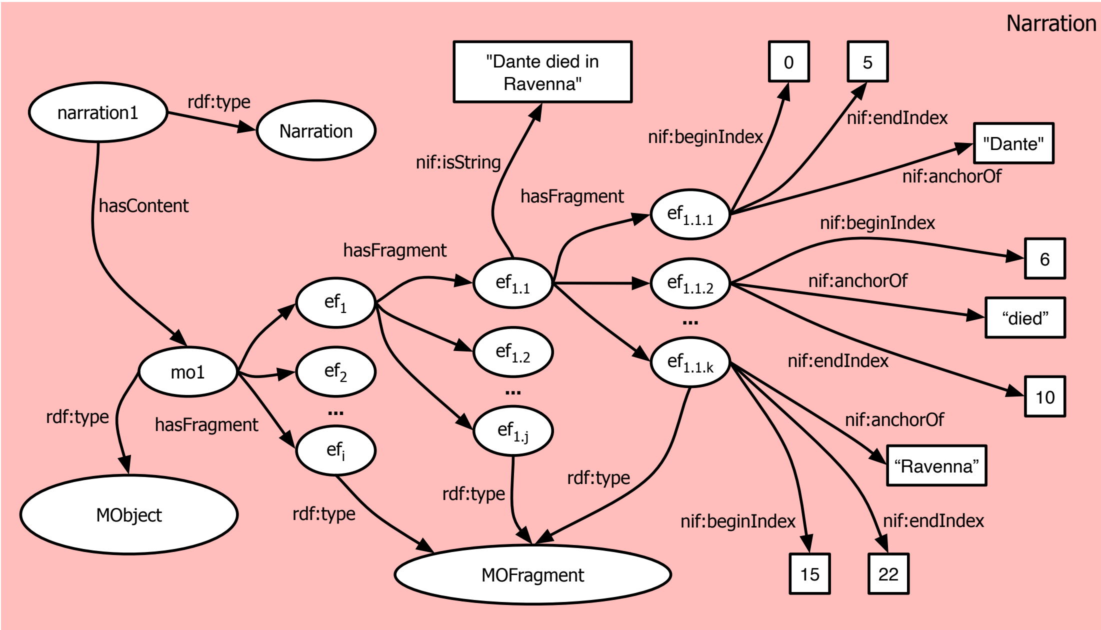

Figure 5.13: *Representation of the narration using NIF*

In order to extract the plot, a machine needs to look at both the ontology graph and the text itself, matching the textual fragments contained in the graph to the full text to understand their order. This order is corresponds to the *textual mereology* defined in Section 4.2.3.

One way to accomplish this goal would be to directly link each textual fragment to the adjacent ones in the text. However, this would need to take into account the possible overlaps between fragments, and the fact that each event could be expressed by multiple disjoint fragments.

For this reason, we propose an extension to the ontology in order to represent the plot more explicitly, by directly expressing the *plot ordering* relation among events. Unfortunately, it is not possible to simply link each pair of events through a direct property, because a fabula may have more than one associated plot. Therefore, it would become impossible to distinguish plots from one another.

We hereby propose a solution that involves a form of *reification*, i.e. each plot is expressed explicitly as a resource in the ontology. Each plot consists of one or more *plot units*, and each plot unit is connected to one (and only one) event, as shown in Figure 5.14.

The *plot ordering* relation is thus expressed not between the events themselves, but between the corresponding plot units.

While this kind of structure may seem redundant, it allows us to better handle multiple plots on the same fabula, even when the narration corresponding to each plot has no associated media object.

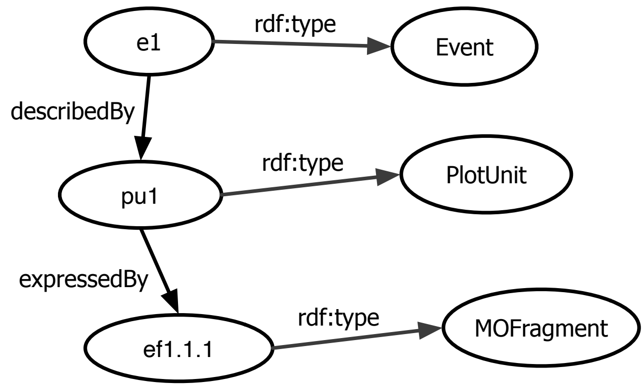

Figure 5.14: *Explicit representation of plot units through reification*

5.5.3 Representing Event Types

In order to create our corpus, we need to adopt a representation of biographical events that allows us to define the *event types*, i.e. the subclasses of the class Event to be annotated in the corpus (e.g. Birth, Death, Travel). This is not a trivial problem to solve, because at present there is no standard model for the representation of biographical events.

By analyzing the projects referenced in Chapter 2, we identified three possible solutions for our needs:

1. *Adopt the Wikidata representation of events*. Given our adoption of Wikidata as reference knowledge base, *prima facie* this seems the easiest solution. However, the Wikidata ontology is very complex and in constant flux. Furthermore, it is also extremely vast, containing thousands of different subclasses of Q1190554 Occurrence (the Wikidata class that is equivalent to our Event).

2. *Adopt an ontology-based representation of events*. In theory, we could adopt a representation of events that is imported from another ontology, but unfortunately we haven't found any ontology that provides a detailed classification of biographical events. Some types of events (e.g. Birth, Death) are defined in the CIDOC CRM, but they are very few and general in nature;

3. *Define our own event types and map them to Wikidata and other ontologies*. This solution is more laborious, but at the same time it gives us more freedom in the definition of event classes, without the need to adopt the whole overly-specific Wikidata model of events. At the same time, we can avoid straying too far from the beaten path by anchoring our representation to existing ontologies through a mapping, whenever possible.

We have decided to adopt this latter solution, for maximum flexibility in the definition of our event classes.

5.5.4 Representing Event Schemas

In many cases, it may be useful to represent descriptions that give the structure of a whole class of events. These descriptions, which we call *event schemas*, represent something that may happen and that has always the same structure at every occurrence.

In an ontology, an event schema consists of a set of axioms that characterize the events that conform to the schema. We will say that an event conforms to the schema if it is structured according to the schema's description.

There are (at least) two approaches to model an event schema:

1. As a sub-class of the class Event. In this approach, an event that conforms to the schema is an instance of the corresponding class, linked to it by property rdf:type.

2. As an individual, carrying the description via the links to appropriate individuals and classes. In this approach, the conformance relation is a property other than rdf:type.

The former approach would seem prone to limit the freedom of representing the schema description by submitting it to the well-defined object-instance relation and to its consequences. However, as we shall see, the choice of rdf:type as conformance relation does not pose any expressive restriction, while resting on a very solid basis. We will therefore adopt it. However, we will continue to use the term event schema instead of event class, for clarity.

To illustrate our ideas, we will use a well-known example from the sport domain by modeling the schema of football matches. To develop the schema, we will use a football match, as shown in Figure 5.15.

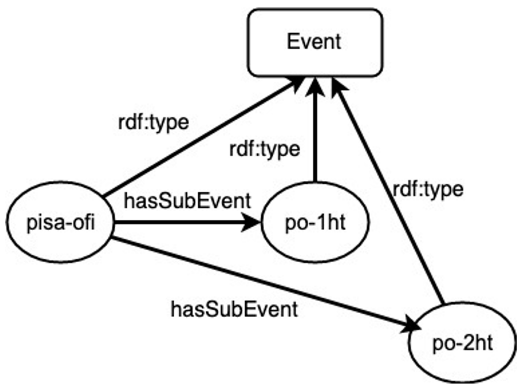

Figure 5.15: *Representation of a football match*

As we have seen, NOnt events span three dimensions: the mereological, the temporal and the causal dimension. As a consequence, an event schema must provide details about how its conforming events behave in each of these dimensions.

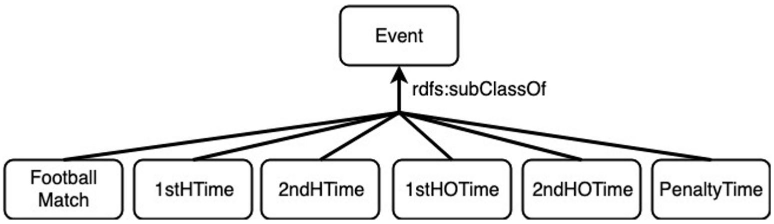

Figure 5.16: *The structure of a football match represented through subclasses of* Event

Mereological Dimension of Event Schemas

From a mereological point of view, our football match event has two parts (see Figure 5.15):

- po-1ht, the event standing for the 1st half-time of the match, and
- po-2ht, the event standing for the 2nd half-time of the match.

The connection between the match event and each of these is clearly a whole-part one, modelled by property hasSubEvent. Now, this structure applies to every football match event, in the sense that a football match event has always two sub-events, the first half and the second half. In addition, some matches may have an overtime, also structured as a first half and a second half. Finally, some matches may include penalties.

Generally speaking, along the mereological dimension a schema S describes the connection between its instances and the events that are parts, or sub-events, of them. For this reason, we will call each of these schemas a sub-schema of the given schema, or the given schema a super-schema of each of them.

To reflect the structure of a football match, then, we define an event class for each phase of a match: 1st-half-time, 2nd-half-time, 1st-half-overtime, 2nd-half-overtime and penalty-time. The instances of these classes capture each the corresponding phase of a football match. In the approach that we are considering all these schemas are sub-classes of Event, as shown in the Figure 5.16.

In general, ff event schema S is a sub-schema of an event schema S′, then for every instance s′ of S′, there exists zero, one or more instances s of S such that s is a sub-event (or a part) of s′.

For instance, since 1st-half-time is a sub-schema of FootballMatch, then for every instance of FootballMatch event fm there exists exactly one instance of 1st-half-time that is a sub-event of fm, and there may exist at most one instance of 1st-half-overtime that is a sub-event of fm.

Notice that Figure 5.16 is logically equivalent to Figure 5.15, as far the three events are concerned, because all these events are instances of class Event, except that in Figure 5.16 they are also instances of the appropriate sub-classes of Event.

We can refine the model shown in the last Figure by using more precise properties to connect an event and its sub-events, namely properties that reflect the specific sub-event in their range. Each one of these properties, which we term *mereological subproperty*:

1. is a subproperty of the event mereological property hasSubevent, so that the structure in the last Figure is preserved;
2. has the super-schema class as domain; and
3. has the corresponding sub-schema class as range.

In our football example, we defined five mereological subproperties of has-SubEvent, one-to-one with the five sub-schemas of FootballMatch. We name these properties by prefixing has to the corresponding event schema class. The domain and range of these properties, as well as their taxonomy, are shown in Figure 5.17 (for readability only the domain of po-1st-half-time is shown in the Figure; as said above, all the other four properties have the same domain).

Figure 5.17 presents the modeling of our example by using the mereological subproperties:

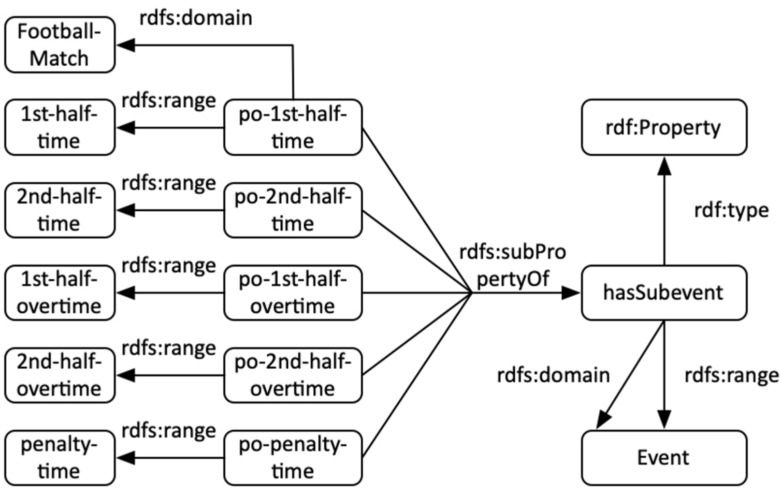

Figure 5.17: *Modeling of event schemas using mereological subproperties*

Again, it is important to note the equivalence of Figure 5.17 with Figure 5.15.

This approach has the advantage of allowing to express any cardinality axiom on a mereological subproperty independently from the other mereological subproperties. For instance, it is possible to state that any instance of Football-Match has exactly one first-half-time and exactly one second half-time, and at most one instance of the other three sub-schemas (using a mixed notation):

```
FootballMatch rdfs:subClassOf
    (= 1 has1stHTime) AND
    (= 1 has2ndHTtime) AND
    ( 1 has1stHOTime) AND
    ( 1 has2ndHOTime) AND
    ( 1 hasPenaltyTime)
```

Notice that the last axioms are not expressible in RDF Schema.

Temporal Dimension of Event Schemas

Along the temporal dimension, we find temporal relations that hold amongst the intervals of the instances of the sub-event schemas of an event schema.

In our football example, these relations specify how the temporal intervals at which the sub-events of a football match occur are related, namely: the interval of occurrence of the first half-time is before that of the second half-time, which is before that of the first-half-overtime and so on.

Any one of these axioms would have the form (variables are universally quantified):

```
rdf:type(x,EC)                  [x is an instance of EC]
AND        rdf:type(x1,EC1)     [x1 is an instance of EC1]
AND        rdf:type(x2,EC2)     [x2 is an instance of EC2]
AND        P1(x,x1)             [x has P1 sub-event x1]
AND        P2(x,x2)             [x has P2 sub-event x2]
AND        hasInterval(x1,y1)   [y1 is the interval of x1]
AND        hasInterval(x2,y2)   [y2 is the interval of x2]
IMPLIES    BTR(y1,y2)           [y1 is BTR y2]
```

where EC is the event super-schema, EC1 and EC2 are the involved event schemas, P1 and P2 are the involved mereological subproperties, and BTR is one of Allen's BTRs.

Note that the first three atoms are redundant, because they are implied by the fourth and fifth atoms and by domain and range constraints. For instance, omitting the redundant atoms:

```
has1stHTime(x,x1)
AND        has2ndHTime(x,x2)
AND        hasInterval(x1,y1)
AND        hasInterval(x2,y2)
IMPLIES    before(y1,y2)
```

We can express these axioms in OWL 2 DL as the complex role inclusion axiom (CRIA):

```
SubObjectPropertyOf(
    ObjectPropertyChain(
        ObjectInverseOf( hasInterval )
        ObjectInverseOf( P1 )
        P2
        hasInterval
    )
    BTR
)
```

In the example, the CRIA would be:

```
SubObjectPropertyOf(
    ObjectPropertyChain(
        ObjectInverseOf( hasInterval )
        ObjectInverseOf( has1stHTime )
        has2ndHTime
        hasInterval
    )
    before
)
```

The effect of this axiom is to generate a *before* assertion every time the event schema is instantiated, and consequently an inconsistency if in the instances the factual knowledge about a specific football match does not conform to it, e.g., the interval of occurrence of a first half of a match overlaps with that of the second half of the same match.

Another kind of temporal information that one may want to associate to an event schema is some quantitative information about the interval at which an event occurs, such as the starting time, the ending time or the duration of that interval. For instance, one may want to say that any instance of class 1st-half-time lasts at least 45 minutes.

In general, we can express this fact as the following axiom schema:

```
SubClassOf(
    ObjectSomeValuesFrom(
        ObjectInverseOf( P1 )
        ObjectSomeValuesFrom( hasInterval DR )
    )
    owl:Nothing
)
```

where DR is the data range defined containing exactly the intervals that are intended.For instance, to express that the 1st-half-time of a FootballMatch must last at least 45 minutes, one can use the following axiom:

```
SubClassOf(
    ObjectSomeValuesFrom(
        ObjectInverseOf( has1stHTime )
        ObjectSomeValuesFrom(
            hasInterval( DataSomeValuesFrom hasLength DR45 )
        )
    )
    owl:Nothing
)
```

where DR45 is a data range obtained from xsd:int by restriction, and including only the numbers between 1 and 45. The axiom reads: “no instance of FootballMatch is connected by has1stHTime to an event which has a length that is a data value smaller than 45 minutes. In sum, no 1st-half-time of a match can last less than 45 minutes.

Causal Dimension of Event Schemas

Along the causal dimension, we find the event schemas whose instances have a causal dependency with the instance of a given schema. In our example, any instance of 1st-half-overtime of a match causes an instance of the 2nd-half-overtime. More precisely, a FootballMatch has a 1st-half-overtime if and only if it has exactly one 2nd-half-overtime.

In OWL 2 DL, this kind of dependency can be expressed as an equivalence axiom of the form:

```
EquivalentClasses(
    ObjectSomeValuesFrom ( P C )
    ObjectSomeValuesFrom ( Q D )
)
```

where P and Q are the involved properties. For instance, the two above example dependencies can be expressed as

```
EquivalentClasses(
    ObjectSomeValuesFrom ( has1stHOTime 1stHOTime )
    ObjectSomeValuesFrom ( has2ndHOTime 2ndHOTime )
)
```

which can be also expressed in the abbreviated form:

```
EquivalentClasses(
    ObjectSomeValuesFrom ( has1stHOTime )
    ObjectSomeValuesFrom ( has2ndHOTime )
)
```

because range constraints make the previous specification redundant. In other words, the football matches that have a first-half-overtime are exactly those that have a second-half-overtime, or equivalently, a football match has a first-half overtime if and only if it has a second-half overtime.

Note that these axioms do not cause an inconsistency if a FootballMatch has a 1stHOTime but not a 2ndHOTime event. The axiom just makes the existence of the latter implicit in that of the former, but it does not require that the individual be explicitly mentioned.

Note also that this kind of dependency does not introduce any isCausedBy property assertion between the involved events. To that effect, we need an axiom capturing the first-order sentence:

```
rdf:type(x,EC)              [x is an instance of EC]
AND    rdf:type(x1,EC1)     [x1 is an instance of EC1]
AND    rdf:type(x2,EC2)     [x2 is an instance of EC2]
AND    P1(x1,x)             [x1 is a P1 sub.event of x]
AND    P2(x2,x)             [x2 is a P2 sub.event of x]
IMPLIES    isCausedBy(x1,x2)
```

where as above the first three atoms are redundant. We can express this axiom as the complex role inclusion axiom (CRIA):

```
SubObjectPropertyOf(
    ObjectPropertyChain( ObjectInverseOf( P1 ) P2 )
    isCausedBy
)
```

For instance, to express explicitly that in a FootballMatch the 1st-half-overtime causes the 2nd-half-overtime, we use the following CRIA:

```
SubObjectPropertyOf(
    ObjectPropertyChain(
        ObjectInverseOf( po-1st-half-time )
        po-2nd-half-time
    )
    isCausedBy
)
```

This concludes our description of the Narrative Ontology and its proposed extensions. In the following chapter, we will discuss the NarraNext tool that we have built to experimentally validate the ontology and investigate narrative extraction from text.

CHAPTER 6

The NarraNext Tool

This chapter presents NarraNext, a semi-automatic tool that is able to extract the main elements of narrative from some natural language text uploaded by the user. The tool further allows the user to create a complete narrative based on a text, using the extracted knowledge to populate the ontology.

NarraNext uses Natural Language Processing techniques, including deep neural networks, to extract elements of narrative from text. The tool is integrated with the Wikidata[1] knowledge base, from which it imports knowledge.

NarraNext can be considered an evolution of the previous NBVT (Narrative Building and Visualising Tool)[2] (Metilli, Bartalesi, & Meghini, 2019b), which has been redesigned and rewritten to focus on semi-automatic narrative extraction from text.

6.1 Background

As we explained in the previous chapters, in our vision, Digital Libraries (DLs) should provide narratives as answers to the queries received from their users. These narratives would be represented as semantic networks of events that connect the digital objects of the DL.

After developing our model of narrative, presented in Chapter 5, the next step is to populate the ontology using narratives. The previous version of the ontology has already been validated using NBVT (Narrative Building and Visualising Tool) (Metilli, Bartalesi, & Meghini, 2019b), a semi-automatic software based on Semantic Web technologies.

[1]https://wikidata.org

[2]https://dlnarratives.eu/tool.html

Using NBVT, we created narratives in four case studies[3], including two biographical narratives are about the life of Dante Alighieri, the major Italian poet of the late Middle Ages, and the life of the Austrian painter Gustav Klimt.

We also performed an experiment to explore the integration of our tool with the Europeana digital library, by linking the narrative about Klimt to the digital objects of Europeana (Meghini et al., 2017).

These efforts were partly semi-automatic, but the user had to create narratives starting from some knowledge imported from Wikidata about the subject of the narrative. There was no facility to import a textual narrative, nor to transform a text into a formal narrative according to our ontology model.

Therefore, the narrative building process was still quite time-consuming, because the user had to create every single event in the *fabula*, and populate it with its related entities and other knowledge (e.g. a title, a description, etc.).

To solve this issue, we developed NarraNext, an evolution of NBVT that allows the creation of narratives starting from a text in natural language. We will use NarraNext to validate the new version of our ontology, using as case study the biography of Dante Alighieri.

It should be noted that we envision NarraNext as a semi-automatic tool with a human in the loop, not as a fully automated solution to the problem of extracting narratives from natural language text.

In general, unlike some researchers have done in the past, we prefer not to call the technology we intend to develop "story understanding" or "narrative understanding", because these terms are ambiguous and may imply that the machine understands the full meaning of the narrative itself.

For this reason, we prefer to use the less ambitious and less ambiguous term "narrative extraction", i.e. a mechanical process of parsing some natural language text, identifying the components of narrative that are present in it. These components are then used to populate our existing ontological model, generating a narrative graph. A human narrator will be able to edit the automatically-built narrative, and the interpretation of the narrative will be left to the final user.

6.2 Requirements

The first step in the development of the NarraNext tool has been the identification of a set of requirements for narrative extraction, which will allow us to validate the effectiveness of the tool by verifying that each requirement is successfully satisfied.

As we discussed in Section 3.3.3, NarraNext has to support some new features with respect to NBVT, in order to allow the representation of the textual structure and textual semantics. In particular, the new requirements that we have identified to support narrative extraction are as follows:

1. *Detecting events in the text*. First of all, events have to be detected in the text. For example, the text "Dante died" expresses the event "Death of Dante".

[3]https://dlnarratives.eu/narratives.html

2. *Classifying events*. Events that are detected in the text need to be classified, i.e. categorized into a set of event classes (for example, the 100 Years War could be classified in a class called *Conflict*).
3. *Detecting event arguments*. We need to extract from the text the entity mentions that act as event arguments, i.e. identify the space, time, and the entities that took part in the event (people, objects, concepts, etc.).
4. *Detecting relations between arguments and events*. We need to identify which entities act as arguments of which events, and if possible classify this relation.
5. *Detecting temporal entities*. Temporal entities such as dates, years, and other indications of time need to be extracted from natural language text.
6. *Normalizing temporal entities*. Temporal entities have to be normalized, i.e. their value needs to be understood to correctly place each event in the narrative's timeline.
7. *Making sense of coreferences found in the text*. It would be useful to understand when the text references an entity without mentioning it explicitly, e.g. through a pronoun.
8. *Linking entities to a reference knowledge base*. It is useful to perform entity linking and event linking to connect them to an external knowledge base.
9. *Linking events to a reference knowledge base*. Similarly to entities, events could also be linked to an existing knowledge base.
10. *Constructing the narrative*. Once we have identified all entities and all events, and linked them to each other, we have to construct the fabula and plot of the narrative.

Before delving into the design and implementation of the NarraNext tool, we will now briefly discuss the selection of a reference knowledge base and the choice of a training corpus.

6.3 Selection of a Reference Knowledge Base

When representing events and entities through Semantic Web technologies, it is good practice to re-use IRIs (Internationalized Resource Identifiers) from existing knowledge bases, when possible (Bizer et al., 2011). In order to import existing IRIs into our ontology and our tool, we investigated three of the most popular knowledge bases: Wikidata, DBpedia (Lehmann et al., 2015), and YAGO (Suchanek et al., 2007).

A recent study has compared the quality of these knowledge bases[4] according to various metrics (Färber et al., 2017). The results of this study highlight

[4]We did not consider the other two knowledge bases analysed in the study (Freebase and OpenCyc) because they were both recently discontinued.

that Wikidata is the top-rated knowledge base on the average of the considered metrics, with especially high scores on trustworthiness, consistency, timeliness, relevancy, and licensing.

On the basis of these results and of an analysis we performed, we chose Wikidata as reference knowledge base, for the following reasons:

- it contains the largest number of entities (currently more than 93 million) when compared to the other general-purpose knowledge bases;
- it is compatible with Semantic Web technologies such as RDF(S), OWL, and SPARQL (Erxleben et al., 2014);
- it is fully multilingual, with more than 39% of the entities having labels in multiple languages;
- it aims to collect not just statements about entities, but also the primary sources behind those statements (referenced statements are currently more than 75%);
- it is fully integrated into Wikipedia and other related projects, such as Wikimedia Commons,[5] from which we can import non-structured data such as text and images;
- it adopts a Creative Commons Zero[6] license, equivalent to the public domain, making it easier to re-use the knowledge contained in it.

Notice that Wikidata also has a significant potential downside when compared to the other knowledge bases, i.e. its open and collaborative nature, similar to that of Wikipedia (Piscopo et al., 2017). The users of Wikidata can freely add and edit knowledge, including the class hierarchy, thus altering the ontology in unpredictable ways. However, both humans and bots frequently check the ontology for errors that can be due to users' mistakes or deliberate acts of "vandalism", and correct them. This explains the high value of trustworthiness assigned to Wikidata by Färber et al. (2017).

At the end of our analysis we decided that the advantages of Wikidata outnumbered its shortcomings, and selected it as reference KB for the tool.

NarraNext leverages the knowledge stored in Wikidata for the following purposes:

- linking entities (and events) found in the text to the corresponding Wikidata entities;
- providing the narrator with an additional list of entities to use in the narrative;
- importing basic events about the subject of the narrative, such as the birth and death of a person.

[5]https://commons.wikimedia.org

[6]https://creativecommons.org/publicdomain/zero/1.0/

In order to extract RDF knowledge to import in our tool, we made use of the Wikidata Query Service[7] (WQS), a SPARQL endpoint that provides full access to the knowledge stored in Wikidata and is constantly updated following each change in the KB.

6.4 Selection of a Reference Corpus

In order to train our extraction components, and evaluate their performance, we need one or more annotated corpora. Given the requirements listed in Section 6.2, we would like to have a training corpus annotated with knowledge about events.

At a minimum, the corpus should report event mentions and event arguments, connecting the arguments to each event. It would also be very valuable to have a corpus that classifies arguments and events in a similar way as we do in our ontology, and especially provides event classes that are commonly found in biographies. Furthermore, entity mentions and coreferences may also be desirable.

We analyzed several corpora containing annotated events, including the ACE 2005 corpus (Walker & Medero, 2005), MEANTIME (Minard et al., 2016), ASTRE (K.-H. Nguyen et al., 2016), and others, to understand whether one of them could be adopted for our purposes. The outcome of this analysis is as follows:

- the ACE 2005 corpus (Walker & Medero, 2005) is one of the main reference corpora for narrative extraction. Unfortunately, some important classes that we need to extract are not present in the corpus (e.g. Creation, Education, Residence, Membership).[8] Furthermore, some of the classes that are present in the corpus are significantly under-represented (e.g. Birth), thus making it difficult to effectively use it as a training dataset. Unfortunately, the corpus is also not available under an open license;

- MEANTIME (Minard et al., 2016) is a corpus extracted from news articles published on WikiNews. It is available under a Creative Commons license, but while it annotates event mentions and relations between them and their arguments, it does not annotate event classes beyond the generic Event;

- the ASTRE corpus (K.-H. Nguyen et al., 2016) also contains a set of annotated articles extracted from WikiNews, however its representation of events is not very detailed, because the aim of the corpus is to be used for schema induction, not for event extraction training.

After performing this analysis, we realized that none of these corpora are apt for our purposes. The closest one is ACE 2005, because it provides at least some of the classes that we need, but it comes with significant drawbacks.

[7]https://query.wikidata.org

[8]The ACE 2005 event classes are listed at: https://www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf

6.4.1 A Corpus of Biographies from Wikipedia

Since we did not find a biographical corpus that met all our requirements, we decided to annotate our own corpus based on biographical narratives found in the English Wikipedia.

Given that we envision an application to the representation of the lives of writers, artists, and other creators whose works may be found in a Digital Library such as Europeana, we decided to select a set of biographies of such people.

In particular, we selected 25 biographies about people who lived in Italy from the Middle Ages to the 19th Century, by paying attention to features of the biography such as the time period, the length of the text, the occupation of the subject of the biography, and gender balance.

The corpus was annotated by two people, who carried out the annotation independently. The two annotations were then merged using a consensus-based approach. The annotation contains the event mentions, the argument mentions, and the relations between events and arguments. For the events, we allowed only spans of length 1, following ACE 2005 (Walker & Medero, 2005)), while for arguments we allowed multi-word spans.

While this small corpus is useful for demonstrating the viability of our approach to narrative extraction, our tool has been designed to be compatible with any other corpus that contains event annotations.

6.5 Architecture

In this section we present our prototype of a software architecture for extracting formal narratives from natural language text, which we call NarraNext. The narrative extraction process is illustrated in Figure 6.1, while the role of NarraNext in our larger architecture for narrative building is shown in Figure 6.2.

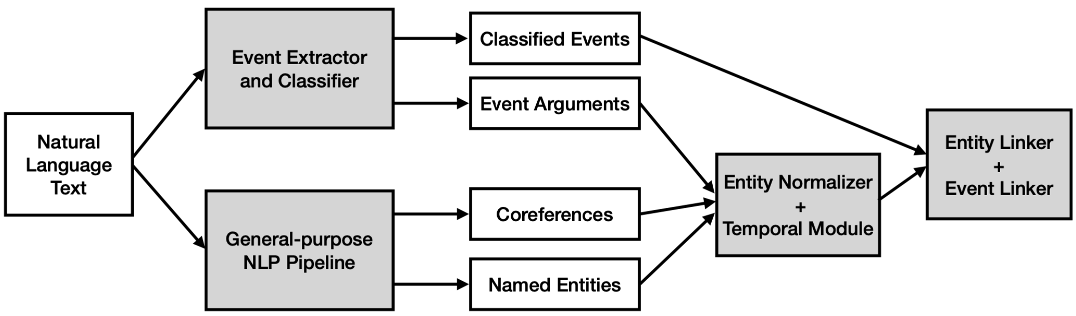

Figure 6.1: *The narrative extraction process of NarraNext.*

The full process to create a narrative in NarraNext is as follows:

1. the user who operates the tool (whom we also call *narrator*) uploads a text in natural language through the Graphical User Interface (GUI) of the tool;

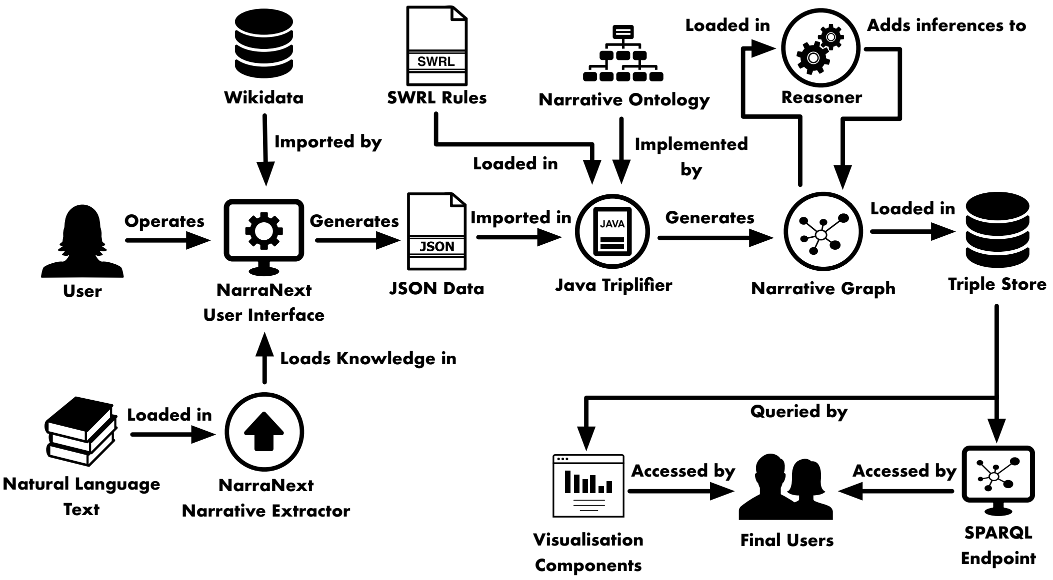

Figure 6.2: *The architecture of NarraNext*

2. the narrative extraction system analyzes the text, extracting the main components of the narrative (events, entities, relations, etc.);

3. the narrator looks at the results of the extraction and, if needed, makes some adjustments;

4. the narrator completes the creation of the narrative through the GUI of the tool, by manually inserting more narrative data based on the narrative ontology and possibly importing resources from Wikidata;

5. the narrative is stored as an intermediate JSON representation;

6. once the narrative is complete, the corresponding JSON representation is given as input to the triplifier, which transforms it into an OWL (Web Ontology Language) ontology encoded as an RDF (Resource Description Framework) graph;

7. A semantic reasoner is used to infer new knowledge, and in particular temporal knowledge;

8. The triplifier stores the resulting graph into a triple store;

9. The final users access the visualization components to explore the completed narrative.

In this section, for each component of the architecture, we briefly discuss the underlying task and we present the technology that we have selected to solve it. A validation of the effectiveness of the architecture is reported in Chapter 7.

Table 6.1: *The components of the NarraNext architecture*

Task	Component	Underlying Technology
Event Detection Event Classification Argument Detection Argument Classification	Event Extractor	DyGIE++
Entity Detection Entity Classification Coreference Resolution	General NLP Pipeline	CoreNLP Stanza
Temporal Entity Detection Temporal Entity Normalization	Time Extractor	HeidelTime
Entity Linking	Entity Linker	OpenTapioca
Event Linking	Event Linker	Custom Component
Intermediate Data Storage	JSON Database	PostgreSQL
Narrative Building	User Interface	Django
Triplification	Triplifier	OWL API
Knowledge Base Storage	Triple Store	Blazegraph

6.6 Implementation

Our narrative extraction architecture is based on a Python[9] backend, built with the Django[10] library. The architecture has been designed in a modular way, where each component can be replaced whenever a better-performing solution is found to solve its underlying task.

The Python backend allowed us to easily integrate NarraNext with the event extraction software that we have developed and re-used, and also with existing Python libraries for NLP. The backend database is PostgreSQL[11], using JSON fields[12] instead of the traditional table-based relational structure.

For the reader's convenience, Table 6.1 reports the full list of tasks that our system is able to handle, and for each one, it reports the name of the component that is assigned the task, and its main underlying technology.

6.6.1 Event Extraction Module

We plan to extract events from natural language text, we have adopted an approach based on deep learning (LeCun et al., 2015). In particular, we have implemented a system to perform the first two tasks that we identified as requirements for narrative extraction, i.e. event extraction and classification.

The *event extractor and classifier* component is able to detect and classify events, which are the central element of our model of narrative. The *event detection* and *event classification* tasks can be performed separately, but recent

[9]https://python.org
[10]https://djangoproject.com
[11]https://www.postgresql.org/
[12]https://www.postgresql.org/docs/current/datatype-json.html

research in the field of narrative extraction has shown that it is more effective to perform the two tasks in a joint way (Cao, 2017).

After studying the main event extraction systems reported in chapter 2, we initially decided to adopt a recurrent neural network architecture based on bidirectional Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997; Schuster & Paliwal, 1997), since the results obtained through this approach had been shown to be particularly apt for this task (Feng et al., 2018).

It should be noted that, in this phase of our research, the goal is not to develop a system with better performance on event extraction from text in comparison with the previous ones, but only to investigate if this approach can be satisfactory in order to be integrated in our architecture to build and visualize narratives.

We started from an open source implementation (Reimers & Gurevych, 2017), which we modified for our purposes, e.g. by adding layer normalization (Ba et al., 2016) [13]). The software takes as input the annotated sentences, tokenizes them, and applies the Komninos dependency-based word embeddings (Komninos & Manandhar, 2016).

The embeddings are given as input to a convolutional neural network (CNN), after which pooling is applied. The results are fed to two LSTM layers of different sizes, in sequence. Dropout is applied both to the CNN and to each LSTM layer. Finally, a hidden dense layer computes the final results by applying a softmax function (Bridle, 1990).

We performed an initial experiment with this software using as training dataset a corpus of 10 biographies from Wikipedia, supplemented with the ACE 2005 corpus (Walker & Medero, 2005). The results were promising (Metilli, Bartalesi, & Meghini, 2019a), but while the integration between the two corpora was cumbersome and difficult to manage, it did not seem to offer a significant advantage

Subsequently, we experimented with an alternative bidirectional LSTM implementation (Reimers & Gurevych, 2019) featuring ELMo embeddings (Peters et al., 2018). We doubled the size of our training corpus of biographies and removed the ACE 2005 corpus, achieving a marginal increase in performance. The structure of this neural network is similar to that of the previous one, except for the embeddings.

Meanwhile, the success of the newer Transformer architecture (Vaswani et al., 2017) and the BERT language representation model (Devlin et al., 2018) forced us to re-evaluate our approach. We selected a neural network system called DyGIE+ (Wadden et al., 2019), which is based on a Transformer architecture and relies on BERT embeddings.

DyGIE++ is able to extract both events and arguments in a single neural network model, with better performance than previous systems. This seemed promising to us, because it has long been known that joint systems provide an advantage in event extraction tasks (Cao, 2017).

[13] An ablation analysis showed that removing layer normalization resulted in a 2.9% decrease in the F_1-score.

Table 6.2: *Comparison of four event extraction solutions*

Event Detection			
Architecture	Precision	Recall	F_1-Score
Bidirectional LSTM with Komninos	75.2	74.3	74.8
Bidirectional LSTM with ELMo	77.2	75.1	76.1
Transformer with BERT-base	80.3	75.0	77.6
Transformer with BERT-large	84.1	75.9	79.8
Event Classification			
Architecture	Precision	Recall	F_1-Score
Bidirectional LSTM with Komninos	74.2	70.6	72.4
Bidirectional LSTM with ELMo	74.0	72.2	73.1
Transformer with BERT-base	78.3	74.6	76.4
Transformer with BERT-large	79.5	76.2	77.8

Furthermore, this software makes it very easy to import pretrained neural network models through the popular Huggingface platform (Wolf et al., 2019). This allows easy testing of different models to identify the best one for a specific task.

The neural network is able to identify *event triggers*, i.e. sequences of words that express the occurrence of an event. The neural network has to be trained using a training corpus. This corpus is highly dependent on the type of text from which the narrative has to be extracted. We tested the three systems on our corpus of biographies from Wikipedia. The results are shown in Table 6.2.

This preliminary testing was performed as follows:

1. The biographies that constitute our test set for the validation experiment in Chapter 7 were held out from the corpus.
2. The corpus was divided into a training dataset comprising 80% of the corpus, and a validation dataset comprising the remaining 20%.
3. We applied k-fold validation by rotating the training dataset and validation dataset 10 times and then averaging the results.
4. We computed the precision, measured as the number of true positives over the total positives, and the recall, measured as the number of true positives over the sum of true positives and false negatives.

Table 6.2 shows the results of the four experiments we ran, reporting the best scores for each experiment. As can be seen, the transformer-based architecture is consistently better than the previous LSTM one, reaching almost 80% F_1-Score in the event detection task and 77% F_1-Score in the event classification task.

Considering the significant advantage given by the transformer-based architecture with BERT-large embeddings, we selected this architecture for our tool.

6.6.2 Event Argument Extraction

The choice of DyGIE++ had the benefit of providing event argument extraction together with event detection. The argument extraction component identifies the arguments, classifies them, and links them to the corresponding events.

Given that the initial results that we obtained with this software looked very promising (especially on event detection and classification), we did not consider a separate event argument extraction solution, or an augmentation of the results of the neural network with a different software.

However, as we will see in Section 7.2.4, we may need to reconsider this choice in the future.

6.6.3 General NLP Pipeline and Entity Normalizer

In addition to the event extractor, we have adopted a *general NLP pipeline* to perform additional extraction tasks. In particular, we use this component for named entity recognition (NER) and coreference resolution.

Theoretically, we could use the DyGIE++ neural network system also to perform these tasks, because it supports them, but since our training dataset does not contain a full annotation of named entities (it is limited to events and event arguments), we prefer to adopt a general pipeline, which also has the benefit of being trained on a larger dataset.

The software library that we have initially selected for this component is Stanza (Qi et al., 2020), which is based on the CoreNLP pipeline (C. Manning et al., 2014). This pipeline has been shown in a recent review to have a higher accuracy than other NLP libraries (Schmitt et al., 2019).

This component is used to provide a classification of all named entities found in the text (not just those that act as event arguments). For what regards the event arguments, the *entity normalizer* component merges the results of the event extractor and the general NLP pipeline using a weighting function that takes into account the confidence intervals reported by each of the two previous components.

By so doing, we are able to suggest to the user the arguments that have a high confidence score and are recognized by both the event extractor and the general NLP pipeline. However, since event argument extraction is a complex problem that is not yet solvable with satisfactory performance, we also propose to the user other named entities based on proximity to the event trigger.

6.6.4 Temporal Module

Temporal extraction is the problem of finding temporal entities mentioned in the text, and recognizing their meaning. For example in the sentence "Dante was born in 1265", the temporal mention is "1265". Sometimes these mentions are explicit dates, or times, but in other cases they are more ambiguous or relative.

For example, it is common to find in a narrative text a sentence of the form "Two years later", making reference to a previous event. It is even more ambiguous when the mention says "some time later". This kind of temporal uncertainty is difficult to handle both in temporal entity extraction and also at the representational level.

Several standards have been developed to annotate temporal mentions, one of the most common ones being TimeML (Pustejovsky et al., 2003).

Our argument extraction component is able to identify temporal arguments, and the general NLP pipeline we have adopted identifies temporal mention using TimeML annotations, but unfortunately neither of these two software is able to correlate a temporal mention with a specific date (or datetime) value (this task is called *normalization*).

Therefore, we looked for a different software that was tailored specifically to temporal entity extraction. Based on a recent survey of such software (Campos et al., 2014), we decided to perform some tests with HeidelTime(Strötgen & Gertz, 2010). This software is able to recognize temporal mentions in the text and also normalize them to obtain their datetime value.

Based on positive initial testing, we decided to select the HeidelTime software for our case study on biographies. This software is at the basis of our *temporal module*.

This module attempts to reconstruct the temporal flow of the narrative, based on the temporal mentions that appear in the text and have been extracted using the previous components.

In the event extractor, we treat the temporal mentions as any other event arguments. If the event has not been successfully connected to a time argument, the temporal module finds the temporal mentions that are closest in the text to the event trigger, among those recognized by the general NLP pipeline, and these are later suggested to the user for addition.

6.6.5 Entity Linker

The last automated step in our narrative extraction process is the *entity linker*, which is used to link entities found in the text, and in particular the event arguments, to the Wikidata knowledge base.

The named entities are first linked to Wikipedia we initially considered the WAT (Piccinno & Ferragina, 2014) entity linking service, a successor of TagMe (Ferragina & Scaiella, 2010) that offers optimal linking to the Wikipedia knowledge base.

The problem of entity linking consists in associating each entity that has been found in the text to a record in an existing knowledge base. In our case, we would like to be able to link both the entities that participate in the events of our narrative (people, places, objects, concepts, etc.), and if possible also the events themselves.

Significant work has been done on linking entities through the development of software such as TagMe and WAT. Generally, these entity linkers connect an entity mention in the text to a page from a single language edition of Wikipedia (e.g. the English one). However, our aim is to connect the entities to Wikidata, which is a much larger multilingual knowledge base.

The problem is that most entity linkers are based on information retrieval techniques that rely on the availability of a long textual description of each entity in the knowledge base. For this reason, they work fairly well with Wikipedia, which contains a textual encyclopedic entry for each of its entities. However, Wikidata is a structured knowledge base and it contains a very limited amount of text compared to Wikipedia.

Therefore, entity linkers that retrieve results from Wikidata often rely on Wikipedia to achieve the linking, and then simply connect the Wikipedia result to the corresponding Wikidata entity. We find this approach unsatisfactory, because it is still limited by the availability of each entity in a single language version of Wikipedia. For example, this kind of entity linker, when trained on the English Wikipedia, will not find an entity which is described only in the Italian Wikipedia, even if both are present in Wikidata.

In order to solve this issue, we have investigated more recent alternative approaches to entity linking. A recent survey about the topic of linking to Wikidata has been performed very recently by Möller et al. (n.d.). Unfortunately, as outlined in the survey, the current landscape of entity linking to Wikidata is very fragmented.

As highlighted by the authors, many disparate datasets have been used for evaluating these tools, and the results are often mixed. Furthermore, a majority of tools are not available under an open license and do not offer public APIs.

After analyzing the available solutions, we decided to investigate the following software:

- OpenTapioca (Delpeuch, 2019) is a lightweight entity linker based on topic similarity that is synchronized in real time with Wikidata. Falcon 2.0 is a rule-based the first joint entity and relation linker based in Wikidata (Sakor et al., 2020);
- Falcon 2.0 is a rule-based the first joint entity and relation linker based on Wikidata (Sakor et al., 2020);
- PNEL (Banerjee et al., 2020) is an entity linker based on the concept of Pointer Networks, and relies on pre-indexed embeddings and entity labels as features for retrieval.

The work of Botha et al. (2020) also seems promising, however it is trained on a very limited dataset from WikiNews, therefore it is not usable for our purposes without some extensive preliminary work.

Based on our analysis and some preliminary testing, we decided to focus on OpenTapioca. This entity linker achieves very good results on some datasets, but average ones on others (Delpeuch, 2019). Yet, it has several interesting advantages when compared to the other solutions:

- it is very flexible, allowing the user to select the full Wikidata graph or only a subset of it to link against, based on SPARQL queries;

- it doesn't require a custom training corpus, therefore it is be general enough to be re-used on different kinds of narrative texts;
- it is very easy to integrate into our tool for testing, requiring only some Python libraries and an Apache Solr[14] backend;
- it is available under an open license.

In particular, the fact that we could directly modify the graph against which to perform the linking looked particularly interesting to us. For example, when creating a biography of a person, the graph could be filtered to exclude all people who lived after the death of that person, and this may improve precision.

Therefore we decided to implement our *entity linker* component based on OpenTapioca. The testing will be performed in Section 7.2.6.

6.6.6 Event Linker

Entity linkers are generally able to link named entities, but they are not able to link events (unless they are nominalized). It would be very useful to connect the events of the narrative to an existing knowledge base.

Unfortunately, most knowledge bases are not event-based. Events such as births and deaths are often represented using simple properties such as *date of birth* or *date of death* that directly connect a person to their date of birth/death, leaving the underlying event implicit. This is also the case on Wikidata, where most event are not represented in an explicit way.

To solve this problem and allow event linking, we generated the Wikidata Event Graph (WEG) that we generated based on the existing Wikidata graph, by making the implicit knowledge about events explicit (Metilli, Bartalesi, Meghini, & Aloia, 2019). Where possible, the events are linked to the WEG through graph matching, taking into account the event time, location, and participating entities.

The Wikidata Event Graph

Ideally, a suitable knowledge base for our purposes should contain: (i) historical entities such as people, places, and physical objects (ii) historical events connecting those entities. However, in most existing knowledge bases, including Wikidata, events are often represented in an implicit way. For instance, the knowledge base currently contains 4.52 million entities about people, but only 6,360 events of type "death", because most people's deaths are expressed implicitly through properties such as P570 date of death.

Many historical events, such as World War II, are represented explicitly in Wikidata, but they make up just 3% of the total number of entities. This is still a significant number overall (1.92 million)[15], but it is not enough for our purposes because in our ontology *all* events are represented explicitly. Indeed, in the four narratives[16] that were developed using our tool, the percentage of

[14] https://solr.apache.org
[15] https://bit.do/ewP9w
[16] https://dlnarratives.eu/narratives.html

events that could be directly matched to Wikidata was less than 2%. This is significantly less than the percentage of related entities that could be found in Wikidata (69%).

In our view, the solution to this problem is the generation of what we call the Wikidata Event Graph (WEG), i.e. the Wikidata graph augmented with an explicit representation of all events implicitly expressed in it. Generating this graph would allow us to reference the events from our ontology and import them into our tool, offering the user a much more complete coverage of historical events.

In order to extract the WEG, we analysed all Wikidata properties[17] and compiled a list of the ones that, in our opinion, express implicit events. For instance, the property P570 date of death expresses an event of type "death".

We developed and implemented an algorithm that allows recognizing Wikidata properties that do not express events. From the current total of 5,234 properties, the algorithm removed several properties based on the following criteria:

1. the type of the property, i.e. meta-properties, properties that connect entities to other Wikimedia projects, obsoleted and deprecated properties, properties classified as "Wikidata property for an identifier", which simply link a Wikidata entity to an identifier in another knowledge base (for instance, the property P214 VIAF ID connecting an individual to its representation in the Virtual International Authority File[18]);

2. the datatype of the property's range literal, i.e. Web pages (datatype URL), numerical quantities (datatype Quantity), geographical features (datatypes GeoShape and GlobeCoordinate), media (datatype CommonsMedia), external IDs (datatype ExternalId), tabular data (datatype TabularData), and Wikidata properties (datatype WikibaseProperty).

At the end of this process, we obtained a list of about 1,000 candidate properties that could potentially express implicit events. Most of these (265) were applied to works, 158 were applied to people, 119 were applied to organizations, and the remaining ones were applied to other types of entities. In order to implement a case study, we decided to focus on the 158 properties about people. This would allow us to use as test case one of the narratives that had been previously constructed with our tool, i.e. the one about the life of Dante Alighieri[19].

To perform our initial experiment, we ordered the list of 158 Wikidata properties obtained in the previous step by usage in the knowledge base. Among the most used, we selected the first 50 that clearly expressed events about a person's life. The list of 50 properties we considered is reported in Table 6.3.

[17]The full list of properties is available at https://www.wikidata.org/wiki/Wikidata:List_of_properties. Another way to explore the properties is the Wikidata Property Explorer, available at https://tools.wmflabs.org/prop-explorer/.

[18]https://viaf.org

[19]https://dlnarratives.eu/timeline/dante.html

Table 6.3: *Wikidata properties expressing implicit events about people's lives.*

Event Type	Property ID	Property Name	Number of Events
Baptism	P1290 P1636	godparent date of baptism	1,840
Birth	P19 P22 P25 P40 P569	place of birth father mother child date of birth	4,519,957
Creation	P50 P57 P58 P61 P84 P86 P87 P110 P161 P170 P178 P800	author director screenwriter discoverer or inventor architect composer librettist illustrator cast member creator developer notable work	546,048
Death	P20 P157 P509 P570 P1196	place of death killed by cause of death date of death manner of death	1,651,865
Education	P69 P184 P185 P512 P802 P812 P1066	educated at doctoral advisor doctoral student academic degree student academic major student of	763,823
Election	P726 P991 P3602	candidate successful candidate candidacy in election	25,775
Foundation	P112	founder	22,359
Marriage	P26	spouse	94,726
Membership	P54 P102 P463	member of sports team member of political party member of	712,299
Occupation	P6 P35 P39 P106 P108 P210 P286 P803 P1075	head of government head of state position held occupation employer party chief representative head coach professorship rector	3,310,621
Residence	P263 P551	official residence residence	60,372

We developed a software that, through the Wikidata Query Service[20], automatically extracts all events that are expressed implicitly by the properties of Table 6.3 from the Wikidata graph. As expected, the number of birth events is the most numerous (4.52 million), since every person has a birth. The second most numerous type of event is occupation (3.31 million), followed by death (1.65 million). The total number of events contained in the subset of the WEG that we generated is 11.71 million, thereby increasing the number of Wikidata events that can be linked from our tool by more than 600%.

The software removes duplicate events, i.e. identical events that can be extracted more than once through multiple properties, by applying the following criteria implemented using rules:

1. if two entities are linked by a direct property and also by its inverse, e.g. the properties *P802 student* and *P1066 student of*, the two events extracted from the properties are merged;
2. if two entities are linked by a symmetric property, e.g. *P26 spouse*, in both directions, the two resulting events are merged;
3. when two properties express the same event, e.g. if the property *P569 date of birth* and the property *P19 place of birth* are applied to the same person, the two resulting events are merged.

In the first version[21] of the narrative of the life of Dante Alighieri constructed using the previous NBVT tool, only one of 53 events (the Battle of Campaldino) was present in Wikidata as instance of the class *Q1190554 Occurrence*. After generating the WEG, we identified in it 31 more events that were present in the narrative. Therefore, the percentage of events in the narrative that could be automatically detected in Wikidata has increased from 1.9% (1 event) to 60.4% (32 events).

Major events such as the birth of Dante, the writing of the *Divine Comedy*, and the election of Pope Boniface VIII are all contained in the WEG, despite not being explicitly present in Wikidata as instances of the class *Q1190554 Occurrence*. Furthermore, the WEG contains 99 more events about Dante's life that are not present in the narrative built using our tool. Many of these are minor events, e.g. the writing of a sonnet, but it can still be useful to propose them to the user during the narrative building process.

6.6.7 Relation Extraction

Relation extraction is the problem of detecting and classifying a relation between two entities found in the text. In our case, the relation between an event and its arguments is based on the type of the argument (see Section 5.4), as follows:

- for time, it is narra:hasTimeSpan
- for places, it is narra:hasPlace

[20]https://query.wikidata.org
[21]https://dlnarratives.eu/timeline/dante.html

- for other types of entities (e.g. people, objects, concepts, works), it is narra:hasParticipant

Therefore, the classification of the relation between an event and its arguments depends on the argument classification task (which can itself be considered a type of relation extraction). The only exception is for people, where it may be useful to extract a more specific relation to identify the role that a person played in the event.

However, there may be many kinds of roles based on the event type, therefore it would be difficult to define a limited set of roles that are general enough to be applied in practice, but also specific enough to be actually useful. For now, we have decided to leave this problem as future work.

For what concerns relations among events, in the ontology we represent three types of relation: temporal, causal, and mereological.

The temporal relations depend on the temporal entities mentioned in the text, therefore their extraction is done through the temporal module that we developed.

The causal and mereological relations are much more difficult to extract, because very often there is no explicit mention of them in the text. Furthermore, the interpretation of causality is quite subjective, and in our definition depends on the narrator's opinion (see Section 4.2.2). We have decided to leave the extraction of these relations as future work.

6.6.8 Triplifier

The triplifier transforms the JSON data into an OWL (Web Ontology Language) knowledge base, encoded as an RDF graph. The organization of the knowledge in the graph follows the structure defined in NOnt. The triplifier uses the OWL API library (Horridge & Bechhofer, 2011) for constructing the OWL graph.

At present, the triplifier is an external Java-based component that can be invoked directly by the user after completing the narrative building process. The user can then download an OWL file containing the narrative graph.

6.6.9 Reasoner

The reasoner is used by the triplifier to infer new knowledge. The triplifier also takes as input the SWRL rules that are used to support reasoning on the narrative. The triplifier uses the Pellet reasoner (Sirin et al., 2007) for performing reasoning on both OWL and SWRL.

In particular, the reasoner is able to apply the SWRL rules defined in Section 5.2.4 for inferring new temporal knowledge based on the partial knowledge that results from the narrative extraction process and/or from user input.

6.6.10 Triple store

After performing the reasoning, the triplifier is able to store the resulting graph into a Blazegraph triple store[22]. We selected this triple store due to its perfor-

[22]https://www.blazegraph.com/

mance and scalability (see (Salehpour & Davis, 2020) for a recent comparison of triple stores).

6.6.11 Visualizations

Finally, the user can access the knowledge stored in the triple store through a Web interface. The knowledge is extracted using SPARQL queries (W3C SPARQL Working Group, 2013) and visualized in several ways, i.e. as a media-rich timeline, as a set of network graphs, and in table format.

6.7 Main Features of the Tool

The user-facing side of NarraNext has been developed using a Python/Django backend and an HTML5/JavaScript frontend with the Bootstrap[23] library.

The tool assists the user in creating the events that compose the narrative, attempting to minimize the cognitive and technical burden in the selection and identification of the involved entities. In the following list, we report the main features of the tool, and each of these is explained through an example extracted from the narrative about the life of Dante Alighieri.

In particular, NarraNext allows its users to:

- Define the name of the event;
- Define the time span of the event;
- Define the place of the event using an appropriate entity from Wikidata or defined by the user;
- Define the entities that participate in an event, e.g. people, objects, concepts, and link them through the appropriate semantic relations. For example, the writing of Epistle VII to Emperor Henry VII by Dante Alighieri is linked to two people (Dante Alighieri and Henry VII) through the property narra:hasParticipant. Furthermore, the event is linked to the place in which it (most likely) occurred, i.e. the Tuscan region of Casentino, through the property hasPlace, and to the time span in which it occurred (17 April 1311) through the property narra:hasTimeSpan.
- Define the type of each event, choosing from a list of predefined options. In the ontology, we represent the types of event using subclasses of narra:Event. For example, the writing of Epistle VII has type narra:Creation, which is equivalent to E65 Creation of the CIDOC CRM.
- Store the textual fragment, if any, providing a narration of the event in natural language. For example, the writing of Epistle VII is described in the narrative by the following textual fragment: "It was during this time that he wrote De Monarchia, proposing a universal monarchy under Henry VII."[24]

[23]https://getbootstrap.com
[24]Text from the English Wikipedia, https://en.wikipedia.org/wiki/Dante_Alighieri

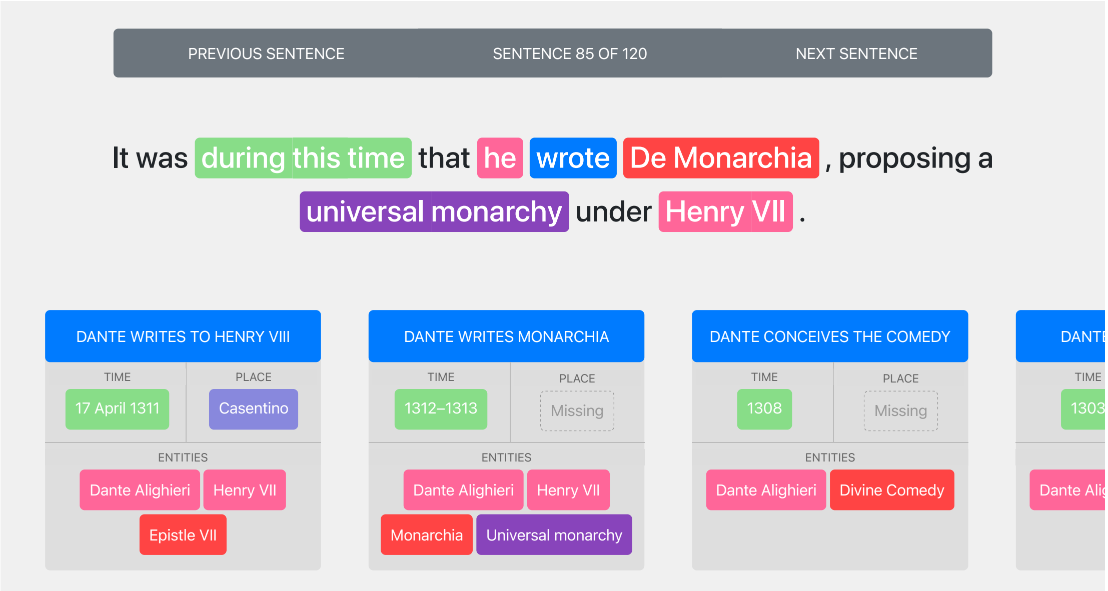

Figure 6.3: *The main interface of NarraNext*

6.8 User Interface

When the user loads the tool for the first time, they are presented with a simple introductory view in which they are asked to upload the text of the narrative (through a text area or by uploading a file), and select the subject of the narrative from Wikidata (through a search field).

The subject of the narrative has to be imported from Wikidata in order to support the loading of related entities and other features (e.g. to aid with automatic entity linking).

the tool runs the narrative extraction components to identify the events and their arguments that are present in the text, and link them to each other and to Wikidata. Then, the extracted knowledge is merged and presented to the user through the *narrative building interface*.

The interface of NarraNext allows the user to build events starting from the ones identified as candidate events by the narrative extraction system.

The interface highlights events, their arguments, and other entities identified by our named entity recognition component. When possible, the named entities are linked to Wikipedia and Wikidata.

The main interface of NarraNext is shown in Figure 6.3. On the top side of the screen, the text of the narrative is presented to the user one sentence at a time. The user can freely move from one sentence to the previous or next one using the buttons at the top, or also jump to another sentence by clicking on the central button that shows the sentence number.

The bottom side of the screen contains the timeline of events. Each event has an editable title, a time, a place, and other entities which are color-coded based on their type. The events are pre-populated based on the results of narrative extraction. Figure 6.4 shows an example of populated event.

Figure 6.4: *The user interface allows the user to create events starting from the text of the narrative.*

It is important to note that the timeline of NarraNext shows the events in the order of the plot, not in the chronological order of the fabula as NBVT does (Metilli, Bartalesi, & Meghini, 2019b). The user can freely change the order of the events through drag-and-drop, thus enabling a more complex manipulation of this aspect of the narrative than is possible in NBVT.

The user can also add, move or delete the entities found in the text (or in other events) through a drag-and-drop functionality, and import other entities from Wikidata, or add user-defined entities, through a separate modal view that appears when clicking on the Entities pane.

To support this functionality, NarraNext performs a set of queries to the Wikidata Query Service[25] to extract from Wikidata all entities related to the subject of the narrative. These queries are run in the background in an asynchronous way, without blocking the user's interaction with the interface.

When the user hovers the mouse on an entity, a popover is shown, describing that entity. For entities found in the text, the popover contains information about the component that extracted the entity, and the reported confidence (if available). For entities linked to Wikidata, the popover also shows some basic information (description, image) imported from Wikidata.

Finally, the user can click on the title of an event to visualize and edit some information that is usually hidden from view to keep the interface as simple as possible. The event is represented as a card that "flips" to reveal these additional fields on its backside. Currently, the user can edit the event type, the text associated to the event, and also link one or more related digital objects.

In addition to the creation of narrative, the tool supports its management as a digital object, enabling the visualization and download of the narrative as a JSON object, and its triplification to obtain an OWL graph.

[25]https://query.wikidata.org

Figure 6.5: *Timeline visualization of the event of Dante's meeting with Beatrice*

NarraNext is currently under active development, therefore it does not yet offer all functionalities that are available in NBVT. However, if the user prefers to use the old NBVT interface, they can also directly import the narrative into NBVT for further editing, given that the intermediate JSON representation is the same (except for some additions, e.g. the confidence level of the annotation, that do not interfere with the tool's functionalities).

6.9 Visualizations

At present, the tool also allows the same visualizations that are accessible with NBVT (Metilli, Bartalesi, & Meghini, 2019b). They are as follows:

- Visualization of the fabula of the narrative on a timeline, including for each event: (i) a title, (ii) a textual description, (iii) the time span of the event, (iv) one or more related entities, and optionally (v) a set of related digital objects, and (vi) an image that illustrates the event.
- Visualising the events that happened in a range of time specified by the user, in form of table, exportable in CSV format.
- Visualising the entities related to a specific event, in graph form.
- Visualising the events related to a specific entity, in graph form.

Figure 6.5 shows an event in the graph of Dante's biography, including the textual description of the event, the related entities, the related digital objects (if available) and an image from Wikimedia Commons.

Figure 6.6 shows a graph visualization of the event of the birth of Dante (left), and a graph visualization of all the events that happened in Florence (right).

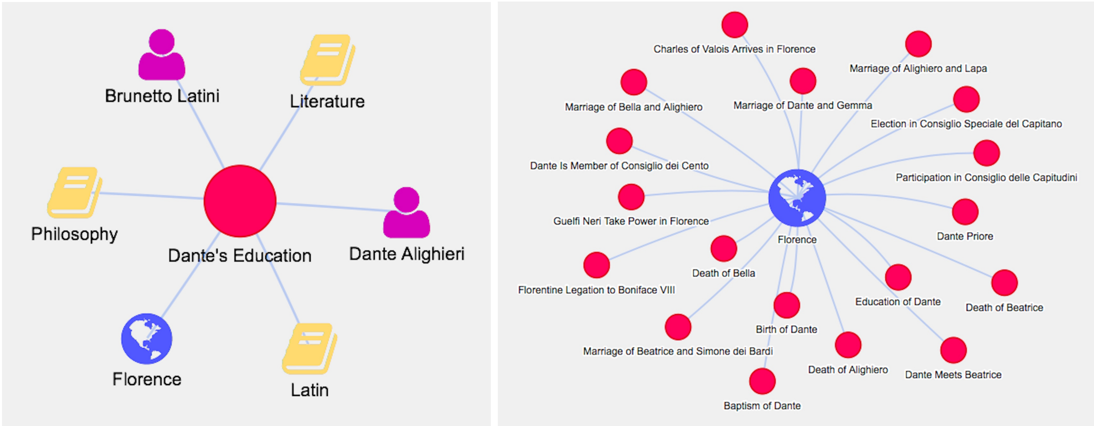

Figure 6.6: *Graph visualization of the event of Dante's education (left) and the events that happened in Florence (right)*

6.10 Enriching the Events of the Narrative Through Europeana

Finally, we are currently working to integrate NarraNext with an external component that provides an integration with the Europeana Digital Library. This component has been presented in (Meghini et al., 2019).

Europeana[26] is the largest European digital library (DL), containing descriptions of more than 58 million cultural heritage objects in various formats such as images, texts, audio, movie files, and 3D objects. The project aggregates metadata from more than 3,500 European cultural institutions, mainly GLAMs (galleries, libraries, archives, and museums.

The Europeana Data Model (EDM)[27], which is compatible with the CIDOC CRM, supports the concept of event, but unfortunately these events do not appear in the Europeana database since they do not show up in the catalogues of cultural institutions from which the descriptions collected by Europeana are imported.

Therefore, we developed a method to suggest to the user a list of digital objects extracted from Europeana that are deemed relevant to a specific event. The objects are ranked according to a score from 1 to 100, where 100 indicates the maximum relevance. The ranking process, which involves queries to the Europeana API and also to Wikidata, is described in detail in (Meghini et al., 2019).

This concludes our description of the NarraNext tool. In the following chapter, we report the results of the experiments that we have ran using the tool in order to evaluate its effectiveness.

[26]https://www.europeana.eu
[27]http://pro.europeana.eu/page/edm-documentation

CHAPTER 7

Experimental Validation

This chapter presents an experimental validation of our work, performed by evaluating the effectiveness of the NarraNext narrative extraction system on a corpus of English-language biographies from Wikipedia.

Biographies are one of the most common kinds of narrative, and they are frequently encountered in our most likely fields of application, i.e. Digital Libraries and Cultural Heritage, therefore they are especially interesting for our purposes.

The simplest way to evaluate the effectiveness of the narrative extraction system would be to take a previously annotated set of biographies, apply our narrative extraction pipeline to the text, and compare the results to the annotation. This kind of evaluation can be performed by using our annotated corpus of 25 biographies from Wikipedia (see Section 6.4). This is the first kind of evaluation that we will perform in the present chapter.

However, the small size of the corpus will only give a partial indication of how effective the narrative extraction system would be on a random biographical text. Furthermore, we have used part of this corpus to train one of the main components of our system (the *event extraction module*, see Section 6.6.1), therefore we cannot use the full corpus for testing.

In order to make sure that the system is general enough to be effective in practice and applicable to different kinds of biographies, it is necessary to widen the dataset on which we can test the system.

For this reason, we have decided to perform two additional evaluations using a random sampling approach, which will allow us to generalize the evaluation to a wider set of biographies. In the following, we will describe the three evaluations and discuss their results.

7.1 Evaluation Methods

To verify the effectiveness of our narrative extraction system and the satisfaction of our requirements, we have performed three separate evaluations, each based on a different set of biographies extracted from the English Wikipedia:

1. a direct evaluation of the system on a fully annotated corpus of 25 biographies;
2. an evaluation on a set of 100 biographies, relying on a sampling approach and a manual annotation of a random subset of sentences extracted from each biography;
3. an evaluation on a wider sample of 1000 non-annotated biographies from the English Wikipedia, relying on the structured data that is present in the Wikipedia page, in order to give a partial estimation of the effectiveness of the narrative extraction system on a larger dataset.

In the first case, we have applied each component of the NarraNext tool in order to extract elements of narrative from the natural language text of the biographies, and computed the results for each component. This evaluation allows us to test the narrative extraction system on a dataset that is particularly relevant for the kind of application that we have in mind, however the significance of the results is limited by the small size of the corpus.

In the second case, we have applied a sampling approach to verify that our results can be generalized to a larger set of biographies, by manually annotating a random sample of sentences from each biography and then evaluating the results. This allows us to give a more general estimate of the effectiveness of the system on a random biographical text.

In the third case, we have performed an evaluation based on the structured data found in each Wikipedia page, using data from the *infobox* that is present in the pages to estimate the effectiveness of the narrative extraction system. This allows us to significantly increase the number of biographies in the set, although the evaluation is less complete because only a subset of narrative elements are expressed in a structured way.

7.2 Evaluation on an Annotated Corpus

Our first evaluation of the narrative extraction system relies on a corpus of 25 biographies from the English Wikipedia that has been annotated following the methodology reported in Section 6.4.

This corpus is composed of biographies of notable Italian writers, artists, and scientists who lived between the 13th century and the 19th century.

The corpus contains a total of 2,131 sentences, or 85.2 sentences per biography. The total number of events is 3,924, and the total number of event arguments is 9,483. The average number of events per sentence is 1.84, while the average number of event arguments per sentence is 4.45. Each event has, on average, 2.41 arguments.

As we reported in Section 6.6.1, 20 of the 25 biographies of this dataset have been used to train a major component of our system, therefore we cannot directly use it to test our full narrative extraction pipeline.

For this reason, we have selected as test set the remaining 5 biographies from the dataset, which have been annotated using the same annotation interface that we developed for the training corpus. This test set (called 5-Bio in the following) has been used to evaluate all the components. Where possible, we will also report the evaluation on the full dataset of 25 biographies (called 25-Bio in the following).

The 5-Bio dataset contains a total of 451 sentences, or 90.2 sentences per biography. The total number of events is 921, and the total number of event arguments is 2,293. The average number of events per sentence is 2.04, while the average number of event arguments per sentence is 5.08. Each event has, on average, 2.49 arguments.

In the following, we report some statistics about the event classes and argument classes of the 5-Bio dataset.

7.2.1 Event Classes

The 5-Bio dataset contains 30 event classes. The distribution of the most common event classes is shown as a pie chart in Figure 7.1.

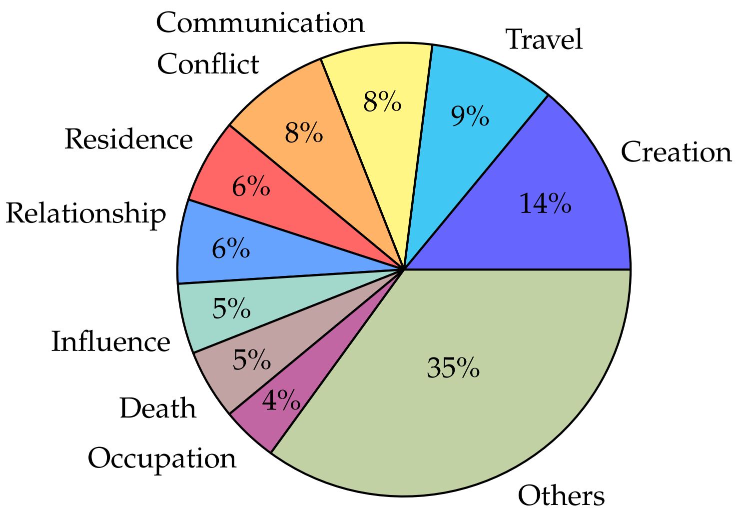

Figure 7.1: *Distribution of event classes in the 5-Bio set*

The full list of event classes is reported in Table 7.1. For each class, we reported a textual example, the number of occurrences and the average number of arguments. Most of the examples refer to the life of the Florentine poet Dante Alighieri, whose biography is part of the dataset.

The most represented event class is Creation (126 instances), reflecting the main activity of the subjects of the biographies that were selected in the corpus. The second most represented class is Travel (85 instances), followed by Communication (73 instances) and Conflict (72 instances).

Table 7.1: *The event classes considered in our case study.*

Event Class	Example	Instances	Avg. Args
Arrest	*They were arrested for treason*	4	1.5
Birth	*Dante was born in Florence*	32	3.7
Burial	*Dante was buried in Ravenna*	3	3.0
Communication	*Dante spoke to Uguccione*	73	2.2
Conflict	*The Guelphs fought the Ghibellines*	72	1.9
Creation	*Dante created the Divine Comedy*	126	3.1
Crime	*They were killed by the Guelphs*	19	1.4
Death	*Beatrice died in 1290*	45	2.3
Destruction	*They destroyed much of the city*	7	1.1
Discovery	*This manuscript was discovered*	7	3.9
Disease	*Dante contracted malaria*	19	1.6
Education	*Dante studied under Brunetto Latini*	31	3.2
Election	*Dante was elected prior*	11	3.5
Foundation	*The city was founded by the Romans*	3	1.7
Influence	*Boccaccio was influenced by Dante*	46	2.5
Marriage	*Dante married Gemma Donati*	20	2.0
Meeting	*Dante met Beatrice Portinari*	25	3.0
Membership	*Dante joined a Florentine guild*	17	2.7
Occupation	*Dante served as prior*	39	2.3
Planning	*Dante planned to return to Florence*	19	2.7
Publishing	*Dante published the book*	12	2.0
Reading	*Dante read Virgil's Aeneid*	4	3.0
Rejection	*Dante refused to participate*	24	2.0
Relationship	*Dante loved Beatrice*	54	2.4
Residence	*Dante settled in Verona*	58	2.4
Sentence	*Dante was condemned to exile*	23	2.3
Transaction	*The book was sold*	26	2.3
Transformation	*The building was transformed into a museum*	12	1.8
Travel	*Dante traveled to Bologna*	85	2.4
Trial	*They were put on trial*	5	1.2
Total		921	2.5

7.2.2 Argument Classes

The event arguments of the 5-Bio dataset are classified in the seven classes that we defined: Time, Place, Person, Organization, Object, Concept, and Work. The distribution of the classes is shown in Figure 7.2.

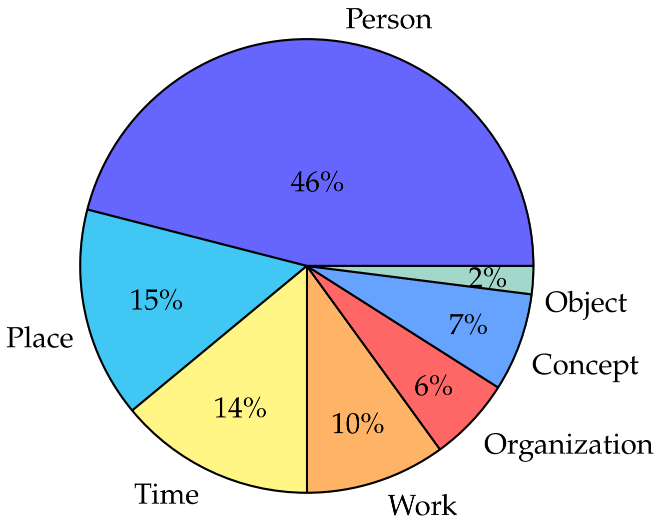

Figure 7.2: *Distribution of argument classes in the 5-Bio set*

Table 7.2 reports the number of occurrences for each argument class. The most represented argument class is Person, with 1,051 instances, distributed over 648 events. It is followed by Place, with 353 instances, then Time, with 320 instances, Organization (146), Work (218), Concept (167), and finally Object, with only 38 instances.

Table 7.2: *The argument classes considered in our case study.*

Argument Class	Example	Instances	Events with Arg.
Time	*Dante died on September 14, 1321*	320	272
Place	*Dante was born in Florence*	353	282
Person	*Dante met Beatrice*	1,051	648
Organization	*Dante supported the White Guelphs*	146	116
Object	*Dante's tomb is in Ravenna*	38	32
Concept	*Dante studied philosophy*	167	129
Work	*Dante wrote the Divine Comedy*	218	152
Total		2,293	854

It is interesting to note that only 272 events (29.5%) have an explicit time associated to them in the text, and only 282 events (30.6%) have an explicit place, but 648 (70.4%) have at least one participating person. In total, 854 events (92.7%) have at least one associated argument of any class.

Organizations are well represented, appearing 146 times, and in 116 events (12.6%). Works are also well represented, with 218 individual instances and 152 events (16.5%) having at least one work. Arguments of class Concept appear 167 times, and in 129 events (14.4%).

Finally, Object arguments are very few (only 38 in 32 events). We have found that individual physical objects are rarely mentioned in this set of biographies, and almost never as arguments of events. However, it is important to note that some arguments that may be interpreted as physical objects (e.g. a letter) may have been classified as instances of the class Work.

7.2.3 Event Detection and Classification

We evaluated the event detection and classification using our *event extraction* component, based on the DyGIE++ architecture (Wadden et al., 2019) and the BERT-large pretrained neural network model (Devlin et al., 2018), which had shown consistently better results in our preliminary testing (see Section 6.6.1). The results of the evaluation on the 5-Bio dataset are reported in Table 7.3.

Table 7.3: *Results of the evaluation of the event extraction task*

Task	Precision	Recall	F_1 Score
Event Detection	81.4	78.1	79.7
Event Classification	75.5	72.2	73.8

Among the total 921 events, our event extraction system is able to correctly detect 719, while the remaining 202 are not detected. This means that the recall is 78.1. Among the 883 detected events, 719 are correct, and the remaining are false positives, therefore the precision is 81.4 and the F_1 score is 79.7.

The performance on this test dataset is slightly lower than the performance on the validation dataset that we reported in Section 6.6.1, but this is generally the case when applying a neural network algorithm on a real-world test dataset.

On the classification task, our event extraction component achieves a precision of 75.5 and a recall of 72.2, both computed as the unweighted macro-average among the values for each event class. The F_1 score is 73.8.

Table 7.4 reports the results for each event class. As can be seen from the table, the false positives and false negatives are not evenly distributed among the event classes.

The event classes that achieve the best F_1 Score are Death (85.1), Birth (86.7), Marriage (83.3), Creation (83.9), and Travel (80.5). These are very common event classes that are well-defined and well-represented in the dataset.

The worst result is for the Burial event class, which is under-represented in the dataset (3 instances). The Reading and Foundation classes also exhibit low scores and are under-represented in the dataset (4 instances and 3 instances, respectively).

Other classes with low scores include those which have a very generic definition and can be expressed in many different ways in the text, such as Planning

Table 7.4: *Event classification results*

Event Class	Precision	Recall	F_1 Score[1]
Arrest	75.0	75.0	75.0
Birth	86.7	81.3	83.9
Burial	50.0	66.7	57.1
Communication	81.8	74.0	77.7
Conflict	84.6	61.1	71.0
Creation	83.9	78.6	81.1
Crime	72.2	68.4	70.3
Death	85.1	88.9	87.0
Destruction	71.4	71.4	71.4
Discovery	80.0	57.1	66.7
Disease	72.2	68.4	70.3
Education	79.3	74.2	76.7
Election	77.8	63.6	70.0
Foundation	66.7	66.7	66.7
Influence	77.8	76.1	76.9
Marriage	83.3	75.0	78.9
Meeting	79.2	76.0	77.6
Membership	75.0	80.0	77.4
Occupation	69.0	74.4	71.6
Planning	75.0	63.2	68.6
Publishing	80.0	66.7	72.7
Reading	60.0	75.0	66.7
Rejection	73.1	79.2	76.0
Relationship	73.1	70.4	71.7
Residence	78.6	75.9	77.2
Sentence	77.3	68.0	72.3
Transaction	77.3	65.4	70.8
Transformation	72.7	66.7	69.6
Travel	80.5	77.6	79.0
Trial	66.7	80.0	72.7
Total	75.5	72.2	73.8

(68.6), Transformation (69.6), Election (70.0), which we also use to indicate appointment to a certain position, Disease (70.3), and Occupation (71.6).

7.2.4 Event Argument Detection and Classification

We evaluated the event argument detection and classification using our *event extraction* component, based on DyGIE++ (Wadden et al., 2019) with BERT-large (Devlin et al., 2018). The arguments were detected and classified using the same neural network architecture that we used for the events, which is able to perform both tasks jointly. The results of the evaluation on the 5-Bio dataset are reported in Table 7.5.

Table 7.5: *Results of the evaluation for the argument extraction task.*

Task	Precision	Recall	F_1 Score
Argument Detection	63.1	54.1	58.3
Argument Classification	61.3	51.7	56.1

Unfortunately, the results on arguments are much lower than those on events. This may be due to the fact that this task can be considered a form of relation extraction, because it does not merely classify the argument itself, but to be successful it also has to recognize that a relation exist between the event trigger and the argument. Moreover, while for the trigger we admit only spans of length 1, for arguments the span length is variable, and this complicates the detection.

One possible workaround is to combine the argument extraction with the named entity recognition that we obtain from our CoreNLP Stanza pipeline, which is much more effective (about 85% F_1 score in our preliminary testing), but is not able to distinguish arguments from non-arguments and doesn't link them to their related events.

Until we find a better solution, it may be preferable to simply propose to the user the named entities found in the text and let them freely choose which entities to set as arguments of the events. The subsequent task that benefits the most from argument extraction (Event Linking, see Section 7.2.7) can then be run after the user selects the correct entities.

7.2.5 Temporal Entity Recognition and Normalization

In this section, we evaluate our *temporal component*, based on HeidelTime (Strötgen & Gertz, 2010), which is used to identify and normalize the temporal entities found in the text.

This is a crucial task because it allows the temporal ordering of events and thus the reconstruction of the fabula of the narrative. Fortunately, the HeidelTime system performs well on our dataset, most likely due to the fact that it has been trained specifically on narrative texts.

This test was performed both on the test dataset of 5 biographies and on the full dataset of 25 biographies. Both results are reported in Table 7.6.

Table 7.6: *Temporal entity recognition and normalization*

Dataset	Task	Precision	Recall	F_1 Score
5-Bio	Temporal Entity Recognition	86.3	77.8	81.8
25-Bio	Temporal Entity Recognition	85.1	74.9	79.7
Dataset	Task	Correct Normalization		
5-Bio	Temporal Entity Normalization	78.4%		
25-Bio	Temporal Entity Normalization	76.5%		

Based on the results of the test, our temporal component is capable of successfully identifying and normalize the vast majority of the temporal entities found in the text (81.8 F_1 Score). This will be very useful for reconstructing the fabula of the narrative.

7.2.6 Entity Linking

After the event arguments have been identified, NarraNext performs *entity linking* to connect them to the corresponding Wikidata entities, using a custom-built OpenTapioca instance (Delpeuch, 2019).

By default, OpenTapioca is able to link only people and places, however, by feeding it the correct SPARQL queries, it is able to import any type of entity from Wikidata, index the entities and successfully retrieve them. As we explained in Section 6.6.5, by constructing suitable SPARQL queries we are able to narrow the amount of irrelevant entities in the index and thus improve retrieval.

For our experiment, we restricted the queries based on time and space. In particular, we filtered by birth date for people, foundation date for organizations, publication date for works, and geographic coordinates for places.

Furthermore, we have modified the software to give a scoring advantage to the entities that are present in the list of entities related to a subject that we import automatically from Wikidata at the beginning of the narrative building process (see Section 6.8).

For example, if the user is working on a biography of Dante Alighieri and the entity linker sees "Beatrice" in the text, it will return several results such as "Beatrice Portinari" (Dante's muse), but also "Beatrice of Silesia" and "Beatrice of Flanders". In this case we give an advantage to Beatrice Portinari, since she appears in our list of entities related to Dante.

Table 7.7 reports the results we obtained by using a default instance of OpenTapioca and our custom version of the software. The percentages in the table do not take into consideration the entities that are missing from Wikidata.

Table 7.7: *Results of the evaluation for the entity linking task.*

Dataset	Task	People	Places	Orgs	Works	Concepts
5-Bio	Entity Linking (default)	85.1%	87.4%	14.9%	—	—
25-Bio	Entity Linking (default)	83.4%	86.2%	13.1%	—	—
5-Bio	Entity Linking (custom)	87.6%	88.9%	66.3%	84.1%	39.8%
25-Bio	Entity Linking (custom)	84.7%	88.2%	64.5%	84.6%	39.2%

As can be seen from the table, we were able to significantly improve the retrieval by using custom queries. We plan to integrate this functionality into NarraNext by preparing some predefined queries that are configured automatically when the user selects the subject of the narrative (see Section 6.8).

7.2.7 Event Linking

After the entities and events have been identified, NarraNext performs *event linking* to connect them to the Wikidata event graph. The linking is done by running SPARQL queries against the Wikidata Event Graph (WEG) we discussed in Section 6.6.6.

Table 7.8 reports the results of the event linking task. Please note that these results refer to the percentage of events that can be successfully linked among those that are actually present in Wikidata (about 40% of the total events).

Table 7.8: *Results of the evaluation for the event linking task*

Dataset	Task	Correct Linking
5-Bio	Event Linking (assuming known time/place)	77.6%
25-Bio	Event Linking (assuming known time/place)	74.9%

This result is satisfactory, and may be further improved by refining our SPARQL queries. The coverage of the Wikidata Event Graph could also be improved to match more events.

However, to achieve this result we rely on the availability of accurate knowledge about the time and place of the events and their related entities. The experiment we ran is based on this assumption, which unfortunately at present cannot be taken for granted (see Section 7.2.4).

Therefore, at the moment this component can be run only *after* the events have been successfully populated with their related entities, either through automated means or by a human user, and the related entities have been linked to Wikidata using our entity linker.

7.2.8 Coreference Resolution

When performing entity linking, we also need to take into consideration coreference resolution. In our test dataset, this is an issue especially for people, because about 31% of them are expressed through coreferences. The number of entities belonging to other classes that are expressed in this way is instead negligible (less than 5% for each class).

We have tested the coreference module of the CoreNLP Stanza pipeline, but our preliminary results achieved only 61.3 F_1 score, which is not satisfactory. We are currently investigating a better solution among the following:

- re-training the CoreNLP coreference component on a different dataset;
- experimenting with the coreference resolution capabilities of DyGIE++, which, however, are much more computationally expensive than the training we already performed;
- adopting a different software, such as SpaCy NeuralCoref.[2]

[2]https://github.com/huggingface/neuralcoref

Until we find a better solution, it may be preferable to offer to the user the results of the current module and let them manually approve the possible coreferences by giving them a list of the most likely candidates.

7.2.9 Discussion of Results

Our experiment on the annotated corpus of 25 biographies from Wikipedia has shown that it is possible to extract a significant amount of narrative elements from text, and use them to reconstruct a significant portion of the narrative.

In particular, NarraNext is able to:

- detect and classify the events with good accuracy
- partly detect the event and classify the event arguments
- make sense of temporal knowledge
- link the entities and the events to Wikidata

The reported results are not yet sufficient to build a narrative in a completely automated way, but given that we involve a human user in the NarraNext tool, we consider them sufficient for our purposes.

The areas that would benefit from significant improvement are the detection and classification of event arguments, and the coreference resolution module. We plan to investigate some better-performing solutions.

In the meantime, we have designed the user interface of the tool to work around the missing data and make it easy for the user to detect and correct possible errors.

7.3 Evaluation Through Random Sampling

In order to verify whether our approach can be generalized to a larger set of biographies, we decided to adopt a random sampling approach on a set of biographies extracted from the English Wikipedia.

In particular, we randomly selected 100 biographies from the English Wikipedia and manually annotated a subset of 5 random sentences for each biography. Then, we evaluated the successful extraction of narrative elements from this set of 500 sentences.

This approach gives us a different perspective on the effectiveness of our narrative extraction system, because this corpus contains sentences from a larger number of biographies, and these biographies describe people with many different backgrounds, nationalities, occupations, etc.

Therefore, this randomly-constructed corpus gives us a better overview of how our narrative extraction system would perform in a real-world setting.

7.3.1 Dataset Selection

Initially, we ran a SPARQL query on the Wikidata Query Service[3] to identify all biographies that are present on the English Wikipedia.[4]

To ensure that all components of our narrative extraction system could be tested properly, we decided to restrict the set to the biographies which fulfill the following requirements:

- the biography describes a real person (i.e. not a fictional character);
- the person lived before January 1, 1900; this choice is due to the fact that our training dataset was trained on biographies of people who lived before the 20th century, therefore it would not be suitable to identify more recent classes of events such as "participation in a television series".

These two conditions were ensured through the following SPARQL query:

```
SELECT DISTINCT ?person ?page
WHERE
{
    # there is a page about a person
    ?page schema:about ?person .

    # the page is on the English Wikipedia
    ?page schema:isPartOf <https://en.wikipedia.org/> .

    # the person is a human
    ?person wdt:P31 wd:Q5 .

    # the person has a date of death
    ?person wdt:P570 ?death .

    # this date is before 1900
    FILTER (?death < "1900-01-01"^^xsd:dateTime)
}
```

At the time of running the experiment, this SPARQL query returned a total of 204,697 biographies. Among this set of Wikipedia pages, we randomly selected 100 biographies.

7.3.2 Evaluation Methodology

The evaluation on the set of 100 biographies has been performed as follows:

1. we split the text of the Wikipedia page into sentences;

[3]https://query.wikidata.org

[4]The two projects are tightly integrated, and the number of biographical pages that are not linked to Wikidata is extremely small. At the time of the experiment, these were only 82 in total (https://en.wikipedia.org/wiki/Category:Articles_without_Wikidata_item).

2. among the set of sentences, we selected 5 sentences at random;
3. we annotated each sentence by hand;
4. the narrative extraction pipeline was applied to the full biography;
5. for each sentence, the result of the manual annotation was compared to the one generated by the narrative extraction pipeline.

Manual Annotation

We extracted the 5 sentences from the text of the Wikipedia page in a random way, and each was annotated using the same methodology that was used for our annotated corpus (see Section 6.4).

Narrative Extraction

After annotating each sentence, we applied our narrative extraction pipeline to the corresponding biography, and compared the results to the manual annotation.

7.3.3 Statistics About the Dataset

The randomly-constructed corpus based on a set of 100 English Wikipedia biographies contains a total of 500 sentences. The total number of events is 820, and the total number of event arguments is 1872.

The average number of events per sentence is 1.64, while the average number of event arguments per sentence is 3.74. Each event has, on average, 2.28 arguments.

These statistics are similar to those found in the test dataset presented in Section 7.2. However, the distribution of event classes is much different, as shown in Figure 7.3.

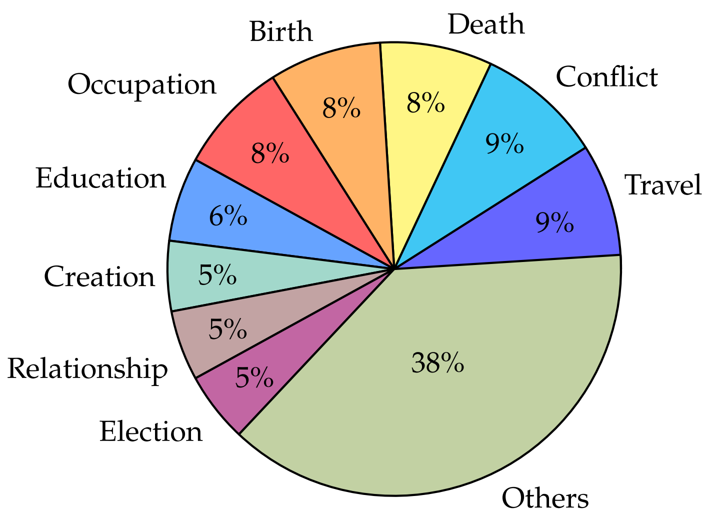

Figure 7.3: *Distribution of event classes in the random sample dataset*

Unlike the distribution of events, the distribution of event arguments is very similar to that found in the test dataset presented in Section 7.2. This can be seen in the pie chart in Figure 7.4.

The only exception is the class Work, which is less present in this dataset due to the fact that the test dataset contained lots of references to creative works made by the writers and artists featured in the dataset.

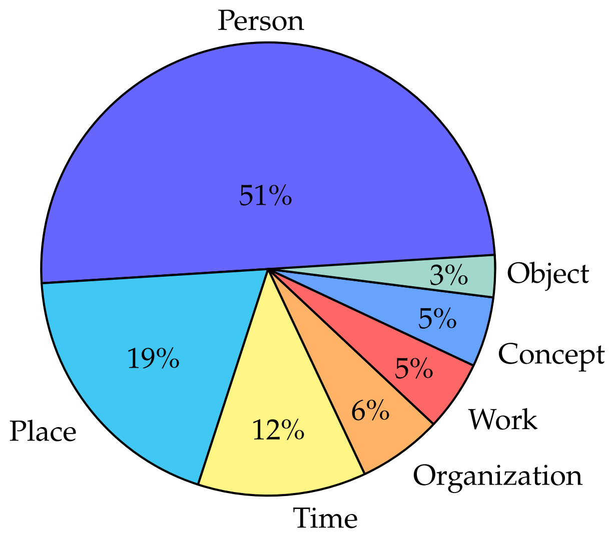

Figure 7.4: *Distribution of argument classes in the random dataset*

7.3.4 Results of the Evaluation

The results of the evaluation are reported in Table 7.9.

Table 7.9: *Results of the random sampling evaluation*

Task	Precision	Recall	F_1 Score
Event Detection	79.9	77.2	78.5
Event Classification	73.2	72.0	72.6
Argument Detection	59.2	51.0	54.8
Argument Classification	55.7	50.3	52.9
Temporal Entity Recognition	84.3	75.2	79.5
Task			Accuracy
Temporal Entity Normalization			76.8%
Task			Accuracy (average)
Entity Linking (custom)			75.7%

The results have been computed individually on each set of sentences, and then averaged over all the sets in the corpus.

7.3.5 Discussion of Results

The results reported in Table 7.9 are slightly lower than those computed on our annotated corpus (see Section 7.2), as expected given the different nature of the dataset and the much different distribution of the event classes.

However, the scores for event detection and classification are still high enough to be usable in a practical application, especially one with a human in the loop, such as our NarraNext tool. A possible improvement could be achieved by training the event extraction component on a larger, more varied dataset.

It should also be noted that the temporal extraction module and the event linking module, which were not trained on our annotated corpus, show similar scores to those of the previous evaluation.

7.4 Evaluation Through Structured Data

In order to verify whether our approach can be further generalized, we decided to investigate the use of structured data that is found in Wikipedia pages to evaluate the effectiveness of our narrative extraction system.

This method of evaluation allows us to increase the size of our test dataset without having to manually annotate a large number of biographies, which requires a costly and time-consuming effort.

In particular, we randomly selected 1,000 biographies from the English Wikipedia and applied our narrative extraction system on them. Then, we verified the satisfaction of our requirements by looking at the structured data that is present in the Wikipedia pages, and verifying whether the narrative components described in them are successfully extracted by our system.

It should be noted that this structured data is not taken as an input by our narrative extraction system, which aims to be applicable to any text in natural language. We simply use these data to evaluate whether the system is effective.

7.4.1 Structured Data in Wikipedia Pages

Wikipedia pages can be considered semi-structured, in the sense that they do not contain only text, but also machine-readable data. This structured data is found in the following sections of each page:

- *infoboxes*, i.e. tables containing various data such as (for a person) birth date and place, death date and place, marriage, offices held, etc. These usually repeat in table form some information that is expressed in the text;
- *internal links* to other Wikipedia pages that are spread throughout the text;
- *tables* that are often located at the end of the page;
- *categories* that describe the topic of the page;
- *link to the Wikidata entity* describing the subject of the page.

Among these types of structured data, the most useful to us are infoboxes, because they effectively contain complete descriptions of events. For example, if the person has a known birth date and birth place, the infobox usually contains this information.

Internal links indicate a general relation of the subject of the page to some other entities, but they do not encode enough information to be useful for our evaluation. Categories may indicate some general aspects of the person's life (e.g. Dante was a poet), but this is too vague to be used effectively. Tables often do not contain descriptions of events, and furthermore they are not in a standard format that can be easily parsed.

The link to the Wikidata entity can be used to access the corresponding Wikidata description, and from there the events (by applying a process similar to the one we followed to build the Wikidata Event Graph, see Section6.6.6). The Wikidata description is fully structured, and already available in RDF format.

The two most promising

However, we cannot *a priori* assume that the infoboxes, or the Wikidata descriptions, contains the same events that are expressed in the text of the related Wikipedia biography.

To verify whether this hypothesis is true, we randomly selected 100 biographies of average length containing infoboxes, and we manually verified how many events found in the infobox are also described in the text of the biography. Then, we accessed the linked Wikidata page and analyzed how many events that are described on that page are also described in the infobox. The results are shown in Table 7.10.

Table 7.10: *Infoboxes vs. Wikidata descriptions*

#	Events in the text	Events not in the text
Infobox	94%	6%
Wikidata description	83%	17%

As shown in the table, the infobox is usually more closely aligned with the text of the page, while the Wikidata description sometimes contains events that are not described in the text of the biography. This is likely due to the multilingual nature of Wikidata, which aggregates events from many language editions of Wikipedia, therefore offering a more complete description of the subject.

Furthermore, it is possible that the Wikidata description of an event differs in some way from the one offered in the text (e.g. a different date or place), because the systems that have been developed to keep them automatically aligned have not been widely deployed in the English Wikipedia.

For the above reasons, it makes more sense for us to use infobox data, instead of Wikidata descriptions, as a proxy for narrative knowledge found in the natural language text.

The main limitation of this evaluation approach is the fact that infoboxes (and, it should be noted, also Wikidata descriptions) usually contain only major events, therefore this kind of evaluation returns only a partial view of the biographical narrative.

For example, the infobox found in the English Wikipedia biography of Dante Alighieri contains the birth and death of the poet, his marriage and his participation in a literary movement (Dolce Stil Novo), but it reports only one notable work (the *Divine Comedy*), omitting any reference to the creation of his several other works.

7.4.2 Dataset Selection

Initially, we ran the same SPARQL query on the Wikidata Query Service[5] that we reported in Section 7.3.1.

However, to ensure that all components of our narrative extraction system could be tested properly, we decided to restrict the set to the biographies which fulfill the following additional requirements:

- the Wikipedia page contains at least one infobox; this is necessary to perform the evaluation;
- the length of the Wikipedia page is at least 2,000 characters; this is due to the fact that very short biographies containing an infobox are less likely to report the contents of the infobox in the text of the biography.

Then, we randomly selected 1,000 Wikipedia pages fulfilling these conditions, and we parsed them to extract the contents of their infoboxes.

Parsing of Wikipedia Pages

The text of each Wikipedia page was extracted using the Wikipedia API. The text is provided by the API in WikiText format, containing markup that identifies structured data such as links, templates, infoboxes, etc. The text of the page was parsed using a Python script that relies on the WikiTextParser library[6].

The infoboxes were identified and extracted using the aforementioned WikiTextParser library. The fields of the infobox were extracted and compared to a list of fields that we have associated to the corresponding biographical events. This list is reported in Table 7.11.

The names of the fields are mostly standardized between different types of infoboxes, but there is some variability, e.g. “place_of_burial” vs. “placeofburial”. For this reason, we matched the strings by removing all spaces and underscores that are found in each field name. Therefore, the list reported in 7.11 is not exhaustive.

From each field, we constructed the corresponding event. For example, from the fields “birth_date = 1265” and “birth_place = [[Florence]]”, we can infer that an event of type Birth occurred in Florence in the year 1265, having as participant the subject of the biography.

[5] https://query.wikidata.org
[6] https://github.com/5j9/wikitextparser

Table 7.11: *Relevant infobox fields*

Event Class	Wikidata Infobox Field
Award	awards
Birth	birth_date
	birth_place
	children
	father
	mother
	parents
Burial	place_of_burial
	resting_place
Conflict	battles
Creation	notableworks
Death	death_date
	death_place
Education	alma_mater
	education
	fields
	school
	students
	teacher
Election	appointed
	appointer
	office
	parliament
	previous_post
	reign
	term_start
	termstart
	title
	titles
Marriage	spouse
Membership	house
	party
	serviceyears
	unit
	teams
Occupation	employer
	occupation
	practice
	profession
	workplaces
	yearsactive
Residence	residence

The value of each field often contains structured data in the form of links or templates that can be parsed. We parsed these structured values in order to reconstruct the biographical events.

In particular, from the links we retrieved the Wikipedia pages about each event argument, and from there the corresponding Wikidata entities. The dates were retrieved from the specific templates, if available, or otherwise identified through a parser.

Narrative Extraction

After retrieving the list of events described in each infobox and parsing them, we applied our narrative extraction pipeline to the corresponding biography, and compared the results to the events retrieved from the infobox.

7.4.3 Results of the Evaluation

We evaluated our narrative extraction system using the infobox-based methodology described above.

This evaluation is quite different from the two previous ones, because instead of extracting the narrative elements that are found in the text and then looking at how many of them were extracted correctly, it instead tells us how many, of the narrative elements described in a specific infobox, are actually present in the text of the biography.

For this reason, we provide the results using a single score that measures the percentage of narrative elements that are successfully matched in the text. This score gives us a general indication of how effective our narrative extraction process is in recovering the main narrative elements found in the text. The results of the evaluation are reported in Table 7.12.

Table 7.12: *Results of the infobox-based evaluation*

Narrative Element	Percentage of matched elements
Event	69.7%
Event Argument	49.3%
Temporal Entity	76.5%
Linked Entity	65.8%

These results cannot directly be compared to the results from the two previous evaluations, however thy give us a general indication of how many of these specific types of narrative elements found in the infoboxes can successfully be matched in the results of the narrative extraction.

The scores have been computed according to the following criteria:

- Event: (i) an event of the same class is found in the text; and (ii) in the same sentence, at least one entity matches the arguments of the event described in the infobox (using either the entity linker or the argument extractor).

- Event Argument: (i) an event argument of this class is found in the text by the argument extractor; and (ii) in the same sentence, an event of the same class as that described in the infobox field is found.
- Temporal Entity: (i) the same temporal entity is found in the text; and (ii) in the same sentence, an event of the same class as that described in the infobox field is found.
- Linked Entity: (i) the same linked entity is found in the text using the entity linker; and (ii) in the same sentence, an event of the same class as that described in the infobox field is found.

7.4.4 Discussion of Results

The results of this evaluation show that a significant percentage of the elements described in the infoboxes can successfully be matched in the text of the corresponding biographies, based on the matching criteria that we reported.

The scores reported in Table 7.12 give us a sense of what percentage of a certain type of narrative element can successfully be matched in a generic biographical text by applying the infobox-based evaluation.

Overall, more than half of events and related entities, and about half of the event arguments, can be successfully matched by our narrative extraction system. These narrative elements can be automatically populated in our narrative building interface, thereby significantly reducing the effort by the human user.

7.5 Satisfaction of Requirements

Our experiments on Wikipedia biographies have shown that it is possible to extract a significant amount of narrative elements from text, and use them to reconstruct a significant portion of the narrative.

The first evaluation showed that NarraNext is able to:

1. detect and classify the events with good accuracy
2. partly detect the event and classify the event arguments
3. make sense of temporal knowledge
4. link the entities and the events to Wikidata

The second evaluation showed that NarraNext is able to perform the above tasks 1–4 on a set of biographies extracted from Wikipedia in a random way. This demonstrates that the narrative extraction system can be successfully applied to different kinds of biographies about different subjects.

The third evaluation showed that NarraNext can successfully match a significant percentage of the narrative elements described in Wikipedia infoboxes, i.e. those pertaining to major biographical events, to their descriptions in the text of the biographies. This demonstrates that the narrative extraction system can be successfully applied on a wider scale to extract narrative elements from a large set of biographical texts.

Table 7.13: *Satisfaction of requirements for narrative extraction*

#	Task	Status
1	Detecting events in the text	Satisfactory
2	Classifying events	Satisfactory
3	Detecting event arguments	Partial
4	Detecting relations between arguments and events	Partial
5	Detecting temporal entities	Satisfactory
6	Normalizing temporal entities	Satisfactory
7	Resolving coreferences	Partial
8	Linking entities to a reference knowledge base	Satisfactory
9	Linking events to a reference knowledge base	Satisfactory
10	Constructing the narrative	Satisfactory

The reported results are not yet sufficient to build a narrative in a completely automated way, but given that we involve a human user in the NarraNext tool, we consider them satisfactory for our purposes.

In Table 7.13, we give an assessment of whether the narrative extraction system's performance is sufficient to fully satisfy the requirements we listed in Section 3.3.3. The last requirement in the list, which has not been tested in this section, is satisfied through the use of our ontology model to transform the extracted data into a semantic network.

The only areas that would benefit from significant improvement are the detection and classification of event arguments, and the coreference resolution module. We hope to find some better-performing solutions, but in the mean time we have designed the user interface of the tool to work around the missing data and make it easy for the user to detect and correct possible errors.

This concludes our experimental validation of NarraNext. In the following chapter, we present two applications of the ontology that serve to further validate it by showing its use in real-world settings. We plan to do the same with the NarraNext tool as future work.

CHAPTER *8*

Additional Applications

This Chapter present two significant additional applications of the ontology, in the following projects: (i) Mingei, a European project about the representation and preservation of Heritage Crafts; (ii) Hypermedia Dante Network, an Italian national research project project aiming to build a Digital Library about Dante's *Divine Comedy*.

8.1 The Mingei European Project

In the last few years, our research group at ISTI-CNR has been a partner of the Mingei European project[1]. This project aims at representing and preserving knowledge about both tangible and intangible aspects of craft as cultural heritage (Zabulis et al., 2020).

Heritage Crafts (HCs) involve craft artefacts, materials, and tools and encompass craftsmanship as a form of Intangible Cultural Heritage (Zabulis et al., 2019). Intangible HC dimensions include dexterity, know-how, and skilled use of tools, as well as tradition and identity of the communities in which they are (or were) practiced. HCs are part of the history of the areas in which they flourish, and also have a significant impact upon their economy.

The project selected three pilot themes that exhibit richness in tangible and intangible dimensions and are directly related to European history: (i) glass, represented by the Conservatoire National des Arts et Métiers (CNAM) in Paris, France, (ii) silk, represented by the Haus der Seidenkultur museum of Krefeld, Germany, and (iii) mastic, represented by the Chios Mastic Museum in Greece.

[1]http://www.mingei-project.eu

Within the Mingei project, our Narrative Ontology has been applied successfully for the representation of narratives about HCs. We report here some details about the project, which constitutes a practical validation of our work.

8.1.1 The Mingei Craft Ontology (CrO)

In Mingei, we developed the Craft Ontology (CrO for short). CrO is an application ontology that uses NOnt as its core ontology. CrO adds to NOnt features that are useful in the context of the Mingei project, such as the notion of process of creation of a craft artefact, and the presentation of a narrative in the context of a museum exhibition.

Moreover, CrO contains specific predicates for the chosen pilots. For instance, it specialises the Ev class into three Ev subclasses, each related to one of the three domains addressed by the project. These subclasses are further specialised on the basis of the domain they represent (e.g. Blowing is a subclass of Glass Activity, that is a subclass of Event).

The added features and predicates are not reported here because they are specifically focused on the Craft Heritage domain, and they cannot be generalised to all possible kinds of narratives. The interested reader may find more details in Zabulis et al. (2020) and Zabulis et al. (2019).

To ensure interoperability, CrO has been integrated with the CIDOC CRM, FRBRoo and OWL Time, relying on the mapping presented in Section 5.4.

In addition, CrO provides three significant extensions to NOnt in order to represent: (i) event schemas (see Section 5.5.4); (ii) the creation process (or recipe) of a HC (see Zabulis et al. (2020)).

8.1.2 The Mingei Online Platform

CrO is currently being populated by the experts of the three pilots using the Mingei Online Platform (Partarakis et al., n.d.), based on ResearchSpace (Oldman & Tanase, 2018). Some of the narratives that are being created based on our ontology are about crafts, i.e. they narrate how craft masters blow glass, weave silk, and process mastic.

In addition to these, there are also narratives that contextualise crafts, concerning for example the history of the town of Krefeld (a center of silk production in Germany), the history of the silk cloth in the Shrine of Charlemagne and its complex motif including the figure of an elephant, and the teaching of glass production at the CNAM in Paris, the history of the association of mastic growers on the island of Chios in Greece.

These narratives will soon be made accessible to scholars and, via special applications, also to the general public visiting the museums involved in the crafts. The enrichment of the narratives through the use of immersive technologies (Partarakis et al., 2020) is also being investigated. An overview of the Mingei approach can be found in Meghini (2020).

8.1.3 Outcomes of the Mingei Project

The main outcome of the Mingei project that is relevant to the present thesis is that the Narrative Ontology presented in Chapter 5, forming the conceptual backbone of CrO, is adequate to support the representational needs of the project.

The scholars who collaborate to the project have particularly appreciated the two levels of representation offered by the ontology, i.e. the fabula and the narration. This double representation allows us to overcome the main limitation of the previous projects addressing the same issue, namely the lack of a strong connection between the documents collected about the domain, which form the narration, and the role that these documents play in the general representation of the craft. Summing up, Mingei proves that narratives are a powerful digitization tool.

In addition to the representation of Craft Heritage, the Mingei ontology has also been applied to the modeling of traditional culinary recipes, which can be viewed as a form of intangible cultural heritage (Partarakis et al., 2021).

This work shows that the Mingei ontology, and the Narrative Ontology it is based on, are powerful and versatile models that can be useful in different contexts.

8.2 The Hypermedia Dante Network Project

The Hypermedia Dante Network Project (HDN) aims to build a Digital Library about Dante's *Divine Comedy*.

The HDN ontology is an evolution of the one developed in DanteSources (Bartalesi, Meghini, Metilli, Andriani, et al., 2017; Bartalesi et al., 2018), integrated with the Narrative Ontology, and having as reference ontologies the CIDOC CRM (Doerr, 2003) and its extension FRBRoo (Riva et al., 2008) (including its in-progress reformulation, LRMoo (Riva & Zumer, 2017)).

The fundamental concept described by the ontology is the *reference*. A reference is an exegetic relation between a text and another entity that is identified by a commentary. A reference can be expressed by the following statement:

A says that knowledge about B can be enriched through C

where A, the source of the reference, is a fragment of text that asserts the reference, and usually comes from a commentary; B, the subject of the reference, is a fragment of text that is clarified by the reference – and in our case study it occurs within a work of Dante's; C is the object of the reference, i.e., the entity that is being referred to.

Within the class of references, we identify three distinct subclasses:

1. *External supports* are references in which the source identifies an object-text that proves useful to support its interpretation of the subject-text.

2. *Loci paralleli* are references in which the source identifies, within an object-text, an idea, a character, or a stylistic feature that, for being used in a

similar way as in the subject-text, helps to clarify it, without necessarily implying a direct derivation.

3. *Citations* are references in which the source establishes that the subject-text directly derives from the object-text. These are clearly the most specific references, and they lie at the center of our ontology.

We identified three characteristics of a reference: the citation type, its content, and the relationship between subject and object. These characteristics always apply to citations, but may not apply to external supports and loci paralleli. We also represent internal references, i.e., quotations to works by the same author of the subject-text. These kinds of references are simply modeled as citations in which the author(s) of the subject-text is the same as the author(s) of the object-text.

8.2.1 The HDN Ontology

The ontology describes three types of citations, which were already identified in the DanteSources ontology: explicit citations (references explicitly made by Dante in his work), strict citations (references to a specific work identified by a scholar), and generic citations (references to a concept put forward by a scholar).

The content of the citation is an explicit or implicit reference to a textual place. We have identified three types and seven sub-types of content:

- *Textual correspondences*: they are references to textual fragments identified by linguistic and stylistic coincidence. These are further divided into: 1. image (the source identifies within the object an image as a precedent for a description or conceptualization made by Dante in the subject), and 2. stylistic feature (the source identifies a linguistic and/or rhetorical precedent for an expression included in the subject-text).
- *Thematic correspondences*: they are references to textual fragments identified by thematic coincidence. These are further divided into: 3. character (the source relates a character mentioned by Dante to a specific object-text), 4. episode (the source identifies a literary episode as an antecedent for Dante's subject-text), and 5. topography (the source believes that a literary memory acts behind Dante's mention of a place).
- *Conceptual correspondences*: they are references to textual fragments identified by conceptual coincidence. These are further subdivided into: 6. motif (the source identifies within a specific object-text the precedent for the development of one of Dante's motifs), and 7. theory (the source identifies within the object-text a theory upon which Dante's text relies).

The relationship between the subject and the object of the citation is classified according to three categories: correction/contradiction, extension/re-elaboration, and confirmation/homology. These three types of relation are inspired by the corresponding properties that are defined in the CiTO ontology (Shotton, 2010).

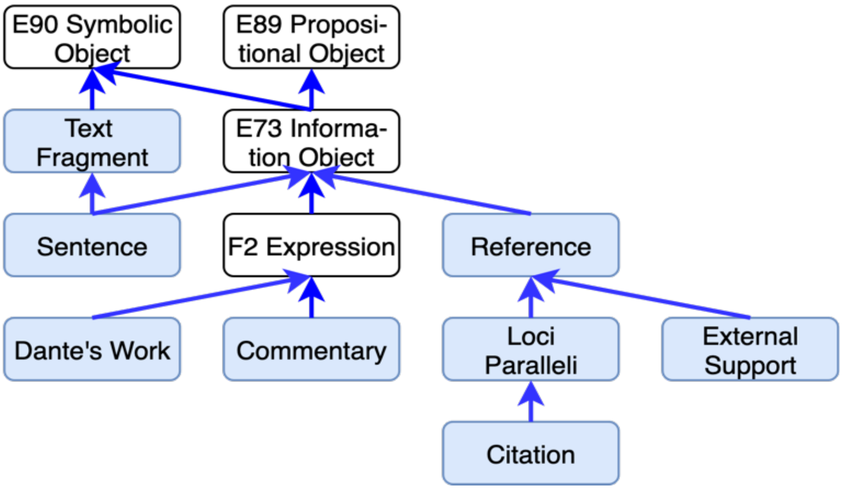

Figure 8.1: *The main classes of the HDN ontology*

8.2.2 Implementation of the ontology

The conceptualization has been formalized in an ontology that has been developed as an extension of the CIDOC CRM and FRBRoo (and its in-progress reformulation LRMoo).

The commentaries to Dante's text are instances of the Commentary class, which is a subclass of F2 Expression. The subparts of the commentaries, including notes, are instances of E90 Symbolic Object. To highlight Dante's works, we introduced the Dante's Work class as a subclass of F2 Expression.

The ontology defines a class for each category of reference, that is:

- Reference is a subclass of E73 Information Object;
- ExternalSupport is a subclass of Reference;
- LociParalleli is a subclass of Reference;
- Citation is a subclass of LociParalleli.

The taxonomy of the classes seen so far is shown in Figure 8.1.

The citation types and subject-to-object relationships are individuals, that is, instances of the E55 Type class. To distinguish them, the ontology includes two subclasses of E55 Type:

1. Citation Type, having as instances genericCitation, strictCitation and explicitCitation;
2. Subject-Object Relationship, having as instances Correction, Extension, and Confirmation.

The subject of the Citation can vary significantly, therefore we consider it as an instance of E1 CRM Entity, the most general class of CRM. An instance of Text Fragment is linked to the part of the work to which it belongs by the FRBRoo property R15i is Fragment Of. A part of the work (e.g. a canto) is linked to the part that includes it (e.g. a cantica) by property P148 is component of.

The references have as elements the source, the subject, and the object. The links between these elements and the references they belong to are represented by the following properties: hasRefSource links a reference to its source, and it is a subproperty of P129i is subject of; hasRefSubject links a reference to its subject, and it is a subproperty of P106 is composed of; hasRefObject links a reference to its object, and it is a subproperty of P129 is about.

8.2.3 Ontology Population Tool

In order to extract the knowledge contained in the commentaries on Dante's Commedia and represent it according to our ontological model, we are developing a tool that will be distributed to scholars to populate the ontology.

The commentaries that will be processed in the HDN project are a selection of the corpus digitized by the Dartmouth Dante Project[2]. The text of the commentaries was pre-processed to identify the notes and, for each note, the following elements: the body of the note, the fragment of Dante's text, the verse, the canto and the cantica.

Once the pre-processing has been performed, scholars will use the tool to populate the ontology with the information that cannot be extracted automatically, that is, the references found in each note and all their characteristics (type, function, content, etc.).

8.2.4 Further Developments of the HDN Project

In the context of the HDN project, two further developments are currently been carried out that will extend the reach of the HDN ontology towards a deeper representation of the textual level of the Comedy.

First of all, an ontology of metaphors is currently been developed, based on work by Tomazzoli (2019). This ontology will allow the representation of metaphors found in the text of the Comedy (Tomazzoli & Meghini, n.d.).

Finally, we are currently working on a new ontology called ORL, which allows the representation of the grammar and syntax of the Italian vernacular of the 14th century, based on previous work by Tavoni (2012) in the context of the DanteSearch[3] project. The ORL ontology is being integrated with NOnt , and also with the Lemon lexical ontology (J. McCrae et al., 2011) to represent the lemmas that are found in the Comedy.

This concludes the chapter about the current real-world applications of the Narrative Ontology. The following chapter reports our conclusions and future works.

[2]https://dante.dartmouth.edu

[3]https://dantesearch.dantenetwork.it

CHAPTER 9

Conclusions and Future Works

The aim of the present thesis has been the enhancement of a current model of computational representation of narrative, with a particular focus on the modeling of the text of the narrative. Furthermore, we have worked towards the extraction of formal narratives from natural language text.

Narratives are a fundamental part of human life. Every human being encounters countless stories during their life, and these stories contribute to form a common understanding of reality. This is reflected in the current digital landscape, and especially on the Web, where narratives are published and shared everyday.

However, the current digital representation of narratives is limited by the fact that each narrative is generally expressed as natural language text or other media, in an unstructured way that is neither standardized nor machine-readable. These limitations hinder the manageability of narratives by automated systems.

One way to solve this problem is the creation of an ontology of narrative, i.e., a formal model of what a narrative is, then develop semi-automated methods to extract narratives from natural language text, and use the extracted data to populate the ontology.

This thesis has attempted to investigate this research question, starting from the state of the art in the fields of Computational Narratology, Semantic Web, and Natural Language Processing. Based on this analysis, we have identified a set of requirements, and we have developed a methodology for our research work.

Then, we have developed an informal conceptualization of narrative, and we have expressed it in a formal way using First-Order Logic. The result of this work is the Narrative Ontology (NOnt), a formal model of narrative that also includes a representation of its textual structure and textual semantics.

To ensure interoperability, the ontology is based on the CIDOC CRM and FRBRoo standards, and it has been expressed using the OWL and SWRL languages of the Semantic Web.

Based on the ontology, we have developed NarraNext, a semi-automatic tool that is able to extract the main elements of narrative from natural language text. The tool allows the user to create a complete narrative based on a text, using the extracted knowledge to populate the ontology. NarraNext is based on recent advancements in the Natural Language Processing field, including deep neural networks, and is integrated with the Wikidata knowledge base.

The validation of our work is being carried out in three different scenarios: (i) a case study on biographies of historical figures found in Wikipedia; (ii) the Mingei project, which applies NOnt to the representation and preservation of Heritage Crafts; (iii) the Hypermedia Dante Network project, where NOnt has been integrated with a citation ontology to represent the content of Dante's Comedy.

All three applications have served to validate the representational adequacy of NOnt and the satisfaction of the requirements we defined. The case study on biographies has also evaluated the effectiveness of the NarraNext tool.

9.1 Main Outcomes of the Thesis

The main outcomes of the present thesis are: (i) the current version of the Narrative Ontology, which has been reworked to better support the textual level of the narrative, and has also been expressed in First Order Logic; (ii) the NarraNext tool, which is able to extract narrative components from text and allows the user to use them as building block to construct a narrative.

9.1.1 Narrative Ontology

We have developed a new version of the Narrative Ontology that is better able to represent the textual and plot levels of narrative, including both structural and semantic aspects of natural language text; this answers research question Q1 by providing solutions for both shortcomings that we had identified in our previous computational representation of narrative.

Furthermore, the ontology has been expressed in First-Order Logic and then through through the OWL and SWRL languages of the Semantic Web. We have also proposed some extensions to better support certain aspects of narratives.

9.1.2 NarraNext Tool

We have developed narrative extraction system that is able to detect narrative elements in natural language text and allows a human user to build a narrative based on them; this answers research question Q2 by providing NarraNext, a comprehensive tool based on several integrated techniques to extract narratives from text in a semi-automatic way.

The tool offers several features that make it more powerful than the previous NBVT, including narrative extraction features but also a more modern user interface (that we plan to test in a future study) and a more reliable backend.

9.1.3 Experimental Validation

We have performed three different evaluations on Wikipedia biographies to verify the effectiveness our narrative extraction system, using as case study a corpus of annotated biographies from Wikipedia. The validation experiment has shown that the NarraNext tool successfully addresses most of the requirements that we had identified.

In addition, we have discussed two further applications of our ontology in the context of Cultural Heritage and Digital Libraries; this answers question Q3 by verifying the effectiveness of the NarraNext tool and building concrete applications based on the Narrative Ontology.

9.2 Future Works

In this section, we discuss the future works that we are considering. Our work is ongoing to further extend the representational coverage of the Narrative Ontology and the efficacy of the NarraNext tool for narrative extraction.

9.2.1 Improvements to Our Work

First of all, several improvements can be made to our current system for narrative extraction. We report a few of them in the following sections.

Narrative Extraction Improvements

As shown in Chapter 7, our narrative extraction system still needs some improvements, especially with regard to the detection and classification of event arguments, and to the coreference resolution. We will attempt to find better solutions to solve these tasks more effectively.

User Testing of the Tool

We need to perform a full user testing of the NarraNext tool in order to better verify the effectiveness and usability of its interface. This will serve to further validate not only the tool, but also the underlying techniques that we have implemented, and indirectly also the ontology itself.

Deeper Integration with Linguistic Ontologies

It would be interesting to continue our study of the representation of linguistic ontologies, and possibly further integrate NOnt with them. Part of this is currently being done in the context of the HDN project, where the ORL ontology is being integrated with both NOnt and Lemon (J. McCrae et al., 2011).

Narrative Comparison and Integration

Once we have extracted a large number of narratives, it will be possible to compare narratives on the same topic or integrate them to create a meta-narrative. This prospect opens up interesting possibilities, especially for the reconstruction of history, where comparing multiple points of view and interpretations of the same historical events could provide a better understanding of them. It is important to note that different narratives on the same subject may contradict each other, therefore an interesting future area of study is the development of methods to resolve such inconsistencies.

9.2.2 Potential Applications

In this section, we discuss some tentative concrete applications that could be built upon our work.

Application to Europeana

We have already worked on the application of our ontology in the field of Digital Libraries, specifically by performing an experiment on Europeana (Meghini et al., 2019). This work could be significantly expanded, adopting Europeana as a testbed for our ontology and narrative extraction tool, for example by building a larger set of curated narratives centered on Europeana digital objects.

Development of Narrapedia

Another application that is enabled by our work is the construction of Narrapedia, i.e. a knowledge base containing a large set of narratives extracted from Wikipedia. This kind of knowledge base would contain complex interrelated narratives that would enable a high-level overview of the historical events described in Wikipedia, and allow the exploration of such events from different points of view. The completion of such work depends on the development of the tools for narrative comparison and integration discussed in Section 9.2.1.

Representation of History

One particularly interesting area of application for our present work is the representation of historical narratives. It is important to note that our current model of narrative, which is strictly event-based, cannot fully account for the nuanced complexities of historical studies. Indeed, history cannot be reduced to a simple network of events; nevertheless, we believe that our work can still provide a useful basis on which to build digital applications to narrate the concrete events that make up history. For this to be accomplished, it will be crucial to collaborate with professional historians.

Educational Applications

Another possible area of application of our work is that of education. As we discussed in Section 3.3.1, NarraNext may be used by a professor as a learning tool. The professor may create a narrative on a topic of study, by importing it from a natural language text, enrich it with some digital objects, and then show it to the students through a timeline visualization. The tool may thus become a powerful learning aid for the students.

Immersive Narratives

Finally, it would be interesting to use the Narrative Ontology and the NarraNext tool to build immersive narratives, through the use of technologies such as virtual reality or augmented reality, drawing on the progress that has been accomplished in the Mingei project (see 8.1). These narratives could be constructed in a more automated way by extracting them from text.

In conclusion, we intend to keep developing both the Narrative Ontology, refining its representational coverage, and the NarraNext tool, to improve its efficacy and accuracy. We hope that our work can be useful, either in practice or as an inspiration, for future developments in the field of computational narratology.

APPENDIX *A*

Ontology Expression in OWL

This appendix contains the expression of the Narrative Ontology through the OWL language of the Semantic Web (W3C OWL Working Group, 2012). The ontology is expressed using the Turtle syntax (Beckett et al., 2014).

The SWRL rules that we have defined to allow temporal reasoning on narratives are available at the following address: https://dlnarratives.eu/ontology/swrl-rules.owl.

A.1 Ontology Expression in OWL

```
@prefix : <https://dlnarratives.eu/ontology#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@base <https://dlnarratives.eu/ontology> .

<https://dlnarratives.eu/ontology> rdf:type owl:Ontology ;
    owl:imports <http://erlangen-crm.org/current/> ,
                <http://erlangen-crm.org/efrbroo/> ,
                <http://www.w3.org/2006/time> ;
    dcterms:creator "Carlo Meghini, Valentina Bartalesi, Daniele
   Metilli" ;
```

```
    dcterms:date "2020-07-22" ;
    dcterms:publisher "ISTI-CNR" ;
    rdfs:comment "An ontology for representing narratives" ;
    rdfs:label "Narrative Ontology" ;
    owl:versionInfo "1.0" .

#################################################################
#    Object Properties
#################################################################

###  https://dlnarratives.eu/ontology#beginsAt
:beginsAt rdf:type owl:ObjectProperty ;
          rdfs:domain <http://erlangen-crm.org/current/E52_Time-
   Span> ;
          rdfs:range :TimePoint ;
          rdfs:comment "This property connects a TimeInterval to
   the TimePoint at which the interval begins."@en ;
          rdfs:label "beginsAt"@en .

###  https://dlnarratives.eu/ontology#endsAt
:endsAt rdf:type owl:ObjectProperty ;
        rdfs:domain <http://erlangen-crm.org/current/E52_Time-
   Span> ;
        rdfs:range :TimePoint ;
        rdfs:comment "This property connects a TimeInterval to
   the TimePoint at which the interval ends."@en ;
        rdfs:label "endsAt"@en .

###  https://dlnarratives.eu/ontology#hadParticipant
:hadParticipant rdf:type owl:ObjectProperty ;
                rdfs:domain :Event ;
                rdfs:range :ActorWithRole ;
                rdfs:comment "This property relates an event with
    an instance of the class ActorWithRole." ;
                rdfs:label "hadParticipant" .

###  https://dlnarratives.eu/ontology#hasContent
:hasContent rdf:type owl:ObjectProperty ;
            rdfs:subPropertyOf <http://erlangen-crm.org/efrbroo/
   R9_is_realised_in> ;
            rdfs:domain :Narration ;
            rdfs:range :MObject ;
            rdfs:comment "This property connects a Narration to
   its content (a MediaObject). Equivalent to the FRBRoo property
    R9 is realised in."@en ;
            rdfs:label "hasContent"@en .
```

```
###  https://dlnarratives.eu/ontology#hasCreator
:hasCreator rdf:type owl:ObjectProperty ;
            rdfs:subPropertyOf owl:topObjectProperty ;
            rdfs:domain <http://erlangen-crm.org/current/
   E18_Physical_Thing> ;
            rdfs:range <http://erlangen-crm.org/current/E39_Actor
   > ;
            rdfs:label "hasCreator"@en .

###  https://dlnarratives.eu/ontology#hasEvent
:hasEvent rdf:type owl:ObjectProperty ;
          rdfs:subPropertyOf <http://erlangen-crm.org/current/
   P9_consists_of> ;
          rdfs:domain :Fabula ;
          rdfs:range :Event ;
          rdfs:comment "This property connects a Fabula to its
   Events. Equivalent to the CRM property P9 consists of."@en ;
          rdfs:label "hasEvent"@en .

###  https://dlnarratives.eu/ontology#hasFabula
:hasFabula rdf:type owl:ObjectProperty ;
           rdfs:subPropertyOf <http://erlangen-crm.org/current/
   P129_is_about> ;
           rdfs:domain :Narrative ;
           rdfs:range :Fabula ;
           rdfs:comment "This property connects a narrative to
   its fabula."@en ;
           rdfs:label "hasFabula"@en .

###  https://dlnarratives.eu/ontology#hasFragment
:hasFragment rdf:type owl:ObjectProperty ;
             rdfs:subPropertyOf <http://erlangen-crm.org/efrbroo/
   R15_has_fragment> ;
             rdfs:domain :MObject ;
             rdfs:range :MOFragment ;
             rdfs:comment "This property connects a media object
   to the fragments that are part of it. Equivalent to the FRBRoo
    property R15 has fragment."@en ;
             rdfs:label "hasFragment"@en .

###  https://dlnarratives.eu/ontology#hasMOType
:hasMOType rdf:type owl:ObjectProperty ;
           rdfs:subPropertyOf <http://erlangen-crm.org/current/
   P2_has_type> ;
```

```
        rdfs:domain <http://erlangen-crm.org/current/
   E1_CRM_Entity> ;
        rdfs:label "hasMOType"@en .

### https://dlnarratives.eu/ontology#hasNarration
:hasNarration rdf:type owl:ObjectProperty ;
              rdfs:subPropertyOf <http://erlangen-crm.org/current
   /P148_has_component> ;
              rdfs:domain :Narrative ;
              rdfs:range :Narration ;
              rdfs:comment "This property connect a narrative to
   the narration that expresses it." ;
              rdfs:label "hasNarration"@en .

### https://dlnarratives.eu/ontology#hasOrderType
:hasOrderType rdf:type owl:ObjectProperty ;
              rdfs:subPropertyOf <http://erlangen-crm.org/current
   /P2_has_type> ;
              rdfs:domain <http://erlangen-crm.org/current/
   E73_Information_Object> ;
              rdfs:label "hasOrderType"@en .

### https://dlnarratives.eu/ontology#hasPlaceIRI
:hasPlaceIRI rdf:type owl:ObjectProperty ;
             rdfs:domain :SpatialRegionStdName ;
             rdfs:label "hasPlaceIRI"@en .

### https://dlnarratives.eu/ontology#hasPresentation
:hasPresentation rdf:type owl:ObjectProperty ;
                 rdfs:subPropertyOf <http://erlangen-crm.org/
   current/P148_has_component> ;
                 rdfs:domain :Narration ;
                 rdfs:label "hasPresentation"@en .

### https://dlnarratives.eu/ontology#hasPresentationSegment
:hasPresentationSegment rdf:type owl:ObjectProperty ;
                        rdfs:subPropertyOf <http://erlangen-crm.
   org/current/P148_has_component> ;
                        rdfs:label "hasPresentationSegment"@en .

### https://dlnarratives.eu/ontology#hasRole
:hasRole rdf:type owl:ObjectProperty ;
         rdfs:domain :ActorWithRole ;
         rdfs:comment "This property connects an ActorWithRole to
    the Role the actor played in the event."@en ;
         rdfs:label "hasRole"@en .
```

```
###  https://dlnarratives.eu/ontology#hasSpatialRegion
:hasSpatialRegion rdf:type owl:ObjectProperty ;
                  rdfs:domain <http://erlangen-crm.org/current/
   E5_Event> ;
                  rdfs:range :SpatialRegion ;
                  rdfs:comment "This property connects an event
   to its spatial region"@en ;
                  rdfs:label "hasSpatialRegion"@en .

###  https://dlnarratives.eu/ontology#hasSubevent
:hasSubevent rdf:type owl:ObjectProperty ;
             rdfs:subPropertyOf <http://erlangen-crm.org/current/
   P9_consists_of> ;
             rdfs:domain :Event ;
             rdfs:range :Event ;
             rdfs:comment "This property connects an Event to
   other Events that are part of it. Equivalent to the CRM
   property P9 consists of."@en ;
             rdfs:label "hasSubevent"@en .

###  https://dlnarratives.eu/ontology#hasSubject
:hasSubject rdf:type owl:ObjectProperty ;
            rdfs:domain :ActorWithRole ;
            rdfs:range <http://erlangen-crm.org/current/E39_Actor
   > ;
            rdfs:comment "This property relates the class
   ActorWithRole with the class E39 Actor." ;
            rdfs:label "hasSubject" .

###  https://dlnarratives.eu/ontology#isCausedBy
:isCausedBy rdf:type owl:ObjectProperty ;
            rdfs:domain :Event ;
            rdfs:range <http://erlangen-crm.org/current/
   E1_CRM_Entity> ;
            rdfs:comment "This property connects an event to
   something that caused it. This property can be used to connect
    events that in normal discourse are predicated to have a
   cause-effect relation, e.g. the eruption of the Vesuvius
   caused the destruction of Pompeii." ;
            rdfs:label "isCausedBy" .

###  https://dlnarratives.eu/ontology#refersTo
:refersTo rdf:type owl:ObjectProperty ;
          rdfs:subPropertyOf <http://erlangen-crm.org/current/
   P129_is_about> ;
```

```
        rdfs:domain :MOFragment ;
        rdfs:range <http://erlangen-crm.org/current/E5_Event> ;
        rdfs:comment "This property connects a media object
   fragment to the event it describes. Equivalent to the CRM
   property P129 is about."@en ;
        rdfs:label "refersTo"@en .

###  https://dlnarratives.eu/ontology#refersToMOFragment
:refersToMOFragment rdf:type owl:ObjectProperty ;
                    rdfs:subPropertyOf <http://erlangen-crm.org/
   current/P129_is_about> ;
                    rdfs:range :MOFragment ;
                    rdfs:comment "This property is used to
   connect a presentation segmnent with the MOFragment it refers
   to" ;
                    rdfs:label "refersToMOFragment"@en .


#################################################################
#    Data properties
#################################################################

###  https://dlnarratives.eu/ontology#hasBoundingBoxMaxLat
:hasBoundingBoxMaxLat rdf:type owl:DatatypeProperty ;
                      rdfs:subPropertyOf owl:topDataProperty ;
                      rdfs:domain :SpatialRegionBBox ;
                      rdfs:range xsd:decimal ;
                      rdfs:label "hasBoundingBoxMaxLat"@en .

###  https://dlnarratives.eu/ontology#hasBoundingBoxMaxLon
:hasBoundingBoxMaxLon rdf:type owl:DatatypeProperty ;
                      rdfs:subPropertyOf owl:topDataProperty ;
                      rdfs:domain :SpatialRegionBBox ;
                      rdfs:range xsd:decimal ;
                      rdfs:label "hasBoundingBoxMaxLon"@en .

###  https://dlnarratives.eu/ontology#hasBoundingBoxMinLat
:hasBoundingBoxMinLat rdf:type owl:DatatypeProperty ;
                      rdfs:subPropertyOf owl:topDataProperty ;
                      rdfs:domain :SpatialRegionBBox ;
                      rdfs:range xsd:decimal ;
                      rdfs:label "hasBoundingBoxMinLat"@en .

###  https://dlnarratives.eu/ontology#hasBoundingBoxMinLon
:hasBoundingBoxMinLon rdf:type owl:DatatypeProperty ;
                      rdfs:subPropertyOf owl:topDataProperty ;
```

```
                      rdfs:domain :SpatialRegionBBox ;
                      rdfs:range xsd:decimal ;
                      rdfs:label "hasBoundingBoxMinLon"@en .

###  https://dlnarratives.eu/ontology#hasCoordinateSystem
:hasCoordinateSystem rdf:type owl:DatatypeProperty ;
                     rdfs:domain :SpatialRegion ;
                     rdfs:range xsd:string ;
                     rdfs:label "hasCoordinateSystem"@en .

###  https://dlnarratives.eu/ontology#hasLatitude
:hasLatitude rdf:type owl:DatatypeProperty ;
             rdfs:subPropertyOf owl:topDataProperty ;
             rdfs:domain :SpatialRegionPoint ;
             rdfs:range xsd:decimal ;
             rdfs:label "hasLatitude"@en .

###  https://dlnarratives.eu/ontology#hasLongitude
:hasLongitude rdf:type owl:DatatypeProperty ;
              rdfs:domain :SpatialRegionPoint ;
              rdfs:range xsd:decimal ;
              rdfs:label "hasLongitude"@en .

###  https://dlnarratives.eu/ontology#hasNumericValue
:hasNumericValue rdf:type owl:DatatypeProperty ;
                 rdfs:subPropertyOf <http://erlangen-crm.org/
   current/P90_has_value> ;
                 rdfs:domain <http://erlangen-crm.org/current/
   E54_Dimension> ;
                 rdfs:range xsd:float ;
                 rdfs:label "hasNumericValue"@en .

###  https://dlnarratives.eu/ontology#hasPlaceName
:hasPlaceName rdf:type owl:DatatypeProperty ;
              rdfs:domain :SpatialRegion ;
              rdfs:range xsd:string ;
              rdfs:label "hasPlaceName"@en .

###  https://dlnarratives.eu/ontology#hasPolygonalRepresentation
:hasPolygonalRepresentation rdf:type owl:DatatypeProperty ;
                            rdfs:subPropertyOf owl:
   topDataProperty ;
                            rdfs:domain :SpatialRegionPolygon ;
                            rdfs:label "
   hasPolygonalRepresentation"@en .
```

```
###  https://dlnarratives.eu/ontology#refersToMOFragmentSource
:refersToMOFragmentSource rdf:type owl:DatatypeProperty ;
                          rdfs:subPropertyOf owl:topDataProperty
   ;
                          rdfs:domain :MOFragment ;
                          rdfs:range xsd:anyURI ;
                          rdfs:label "refersToMOFragmentSource"
   @en .

###  https://dlnarratives.eu/ontology#refersToMOSource
:refersToMOSource rdf:type owl:DatatypeProperty ;
                  rdfs:subPropertyOf owl:topDataProperty ;
                  rdfs:domain :MObject ;
                  rdfs:range xsd:anyURI ;
                  rdfs:label "refersToMOSource"@en .


#################################################################
#    Classes
#################################################################

###  https://dlnarratives.eu/ontology#ActorWithRole
:ActorWithRole rdf:type owl:Class ;
              rdfs:comment """This reification class was
   introduced to allow the assignment of a role to an actor. It
   represents the participation of the actor in the event.

This class has to be linked to:
 an instance of the class E5 Event, through the property
   hadParticipant
 an instance of the class E39 Actor, through the property
   hasSubject
 an instance of the class Role, through the property hasRole""" ;
              rdfs:label "ActorWithRole" .

###  https://dlnarratives.eu/ontology#Event
:Event rdf:type owl:Class ;
       rdfs:comment "This class represents an event. Equivalent
   to the CRM class E5 Event."@en ;
       rdfs:label "Event"@en .

###  https://dlnarratives.eu/ontology#Fabula
:Fabula rdf:type owl:Class ;
        rdfs:subClassOf <http://erlangen-crm.org/current/
   E4_Period> ;
```

```
        rdfs:comment "This class represents the fabula of a
   narrative, i.e. the sequence of events in chronological order
   ." ;
        rdfs:label "Fabula" .

###  https://dlnarratives.eu/ontology#MOFragment
:MOFragment rdf:type owl:Class ;
            rdfs:subClassOf <http://erlangen-crm.org/current/
   E90_Symbolic_Object> ;
            rdfs:comment "This class represents a fragment of a
   media object, i.e. a piece of text, video, audio or other
   media that is a subpart of the media object."@en ;
            rdfs:label "MOFragment"@en .

###  https://dlnarratives.eu/ontology#MOType
:MOType rdf:type owl:Class ;
        rdfs:subClassOf <http://erlangen-crm.org/current/E55_Type
   > ;
        rdfs:label "MOType"@en .

###  https://dlnarratives.eu/ontology#MObject
:MObject rdf:type owl:Class ;
         rdfs:subClassOf <http://erlangen-crm.org/current/
   F22_Self-Contained_Expression> ;
         rdfs:comment "This class represents a media object, i.e.
    a self-contained expression that expresses the narration of a
    narrative. Media objects include texts, videos, audios, etc."
   @en ;
         rdfs:label "MObject"@en .

###  https://dlnarratives.eu/ontology#Narration
:Narration rdf:type owl:Class ;
           rdfs:subClassOf <http://erlangen-crm.org/efrbroo/
   F14_Individual_Work> ;
           rdfs:comment "This class represents the narration of a
    narrative, i.e. an individual work that tells the events of
   the narrative through some form of media (text, video, audio,
   etc.)."@en ;
           rdfs:label "Narration"@en .

###  https://dlnarratives.eu/ontology#Narrative
:Narrative rdf:type owl:Class ;
           rdfs:subClassOf <http://erlangen-crm.org/current/
   E73_Information_Object> ;
           rdfs:comment "This class represents a narrative." ;
           rdfs:label "Narrative" .
```

```
###  https://dlnarratives.eu/ontology#Person
:Person rdf:type owl:Class ;
        rdfs:comment "This class represents a person. Equivalent
   to the CRM class E21 Person."@en ;
        rdfs:label "Person"@en .

###  https://dlnarratives.eu/ontology#Schema
:Schema rdf:type owl:Class ;
        rdfs:subClassOf <http://erlangen-crm.org/current/
   E29_Design_or_Procedure> ;
        rdfs:label "Schema"@en .

###  https://dlnarratives.eu/ontology#Schema_Step
:Schema_Step rdf:type owl:Class ;
             rdfs:subClassOf <http://erlangen-crm.org/current/
   E29_Design_or_Procedure> ;
             rdfs:label "Schema_Step"@en .

###  https://dlnarratives.eu/ontology#SpatialRegion
:SpatialRegion rdf:type owl:Class ;
               rdfs:subClassOf <http://erlangen-crm.org/current/
   E53_Place> ;
               rdfs:label "SpatialRegion"@en .

###  https://dlnarratives.eu/ontology#SpatialRegionBBox
:SpatialRegionBBox rdf:type owl:Class ;
                   rdfs:subClassOf :SpatialRegion ;
                   rdfs:label "SpatialRegionBBox"@en .

###  https://dlnarratives.eu/ontology#SpatialRegionPoint
:SpatialRegionPoint rdf:type owl:Class ;
                    rdfs:subClassOf :SpatialRegion ;
                    rdfs:label "SpatialRegionPoint"@en .

###  https://dlnarratives.eu/ontology#SpatialRegionPolygon
:SpatialRegionPolygon rdf:type owl:Class ;
                      rdfs:subClassOf :SpatialRegion ;
                      rdfs:label "SpatialRegionPolygon"@en .

###  https://dlnarratives.eu/ontology#SpatialRegionStdName
:SpatialRegionStdName rdf:type owl:Class ;
                      rdfs:subClassOf :SpatialRegion ;
                      rdfs:label "SpatialRegionStdName"@en .

###  https://dlnarratives.eu/ontology#TimeInterval
```

```
:TimeInterval rdf:type owl:Class ;
              owl:equivalentClass <https://www.w3.org/2006/time#
   ProperInterval> ;
              rdfs:comment "This class represents a time interval
   . Equivalent to the CRM class E52 Time-Span and to the OWL
   Time class ProperInterval."@en ;
              rdfs:label "TimeInterval"@en .

###  https://dlnarratives.eu/ontology#TimePoint
:TimePoint rdf:type owl:Class ;
           rdfs:comment "This class represents a time point.
   Equivalent to the OWL Time class Instant."@en ;
           rdfs:label "TimePoint"@en .

###  https://www.w3.org/2006/time#ProperInterval
<https://www.w3.org/2006/time#ProperInterval> rdf:type owl:Class
   .
```

Bibliography

Abitebul, S., Hull, R., & Vianu, V. (1995). *Foundations of databases*. Addison-Wesley, Reading, MA. ISBN: 978-0-201-53771-0.

Aguilar, J., Beller, C., McNamee, P., Van Durme, B., Strassel, S., Song, Z., & Ellis, J. (2014). A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. *Proceedings of the 2nd Workshop on Events: Definition, detection, coreference, and representation*, 45–53. https://doi.org/10.3115/v1/W14-2907

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, *26*(11), 832–843. https://doi.org/10.1145/182.358434

Antoniou, G., & van Harmelen, F. (2004). *A Semantic Web primer*. The MIT Press. ISBN: 978-0-262-01242-3.

Aristotle. (2013). *Poetics* (A. Kenny, Trans.). OUP Oxford. ISBN: 978-0-19-163580-9.

Arp, R., Smith, B., & Spear, A. D. (2015). *Building ontologies with Basic Formal Ontology*. MIT Press. ISBN: 978-0-262-52781-1.

Aylett, R. (1999). Narrative in virtual environments: Towards emergent narrative. *Proceedings of the AAAI Fall Symposium on Narrative Intelligence*, 83–86. https://www.aaai.org/Papers/Symposia/Fall/1999/FS-99-01/FS99-01-014.pdf

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv:1607.06450*. https://arxiv.org/abs/1607.06450

Bailey, P. (1999). Searching for storiness: Story-generation from a reader's perspective. *Working notes of the Narrative Intelligence Symposium*, 157–164. https://www.cs.cmu.edu/afs/cs/user/michaelm/www/nidocs/Bailey.pdf

Bal, M. (1997). *Narratology: Introduction to the theory of narrative*. University of Toronto Press, Toronto. ISBN: 978-0-8020-9631-9.

Banerjee, D., Chaudhuri, D., Dubey, M., & Lehmann, J. (2020). PNEL: Pointer network based end-to-end entity linking over knowledge graphs. *The Semantic Web – ISWC 2020*, 21–38. ISBN: 978-3-030-62419-4.

Barabucci, G., Di Iorio, A., Peroni, S., Poggi, F., & Vitali, F. (2013). Annotations with EARMARK in practice: A fairy tale. *Proceedings of the 1st International Workshop on Collaborative Annotations in Shared Environment: metadata, vocabularies and techniques in the Digital Humanities*, 1–8. https://doi.org/10.1145/2517978.2517990

Bartalesi, V., Meghini, C., & Metilli, D. (2016). Steps towards a formal ontology of narratives based on narratology. In B. Miller, A. Lieto, R. Ronfard, S. G. Ware, & M. A. Finlayson (Eds.), *7th workshop on Computational Models of Narrative (CMN 2016)* (4:1–4:10). Schloss Dagstuhl. https://doi.org/10.4230/OASIcs.CMN.2016.4

Bartalesi, V., Meghini, C., & Metilli, D. (2017). A conceptualisation of narratives and its expression in the CRM. *International Journal of Metadata, Semantics and Ontologies*, *12*(1), 35–46. https://doi.org/10.1504/IJMSO.2017.087692

Bartalesi, V., Meghini, C., Metilli, D., Andriani, P., & Tavoni, M. (2017). DanteSources: A digital library for studying Dante Alighieri's primary sources. *Umanistica Digitale*, (1). https://doi.org/10.6092/issn.2532-8816/7250

Bartalesi, V., Meghini, C., Metilli, D., Tavoni, M., & Andriani, P. (2018). A web application for exploring primary sources: The DanteSources case study. *Digital Scholarship in the Humanities*, *33*(4), 705–723.

Barthes, R. (1966). Introduction à l'analyse structurale des récits. *Communications*, *8*(1). https://doi.org/10.3406/comm.1966.1113

Batsakis, S., Petrakis, E. G., Tachmazidis, I., & Antoniou, G. (2017). Temporal representation and reasoning in OWL 2. *Semantic Web*, *8*(6), 981–1000. https://doi.org/10.3233/SW-160248

Becker, H., Naaman, M., & Gravano, L. (2011). Beyond trending topics: Real-world event identification on Twitter. *Proceedings of ICWSM*, *11*(2011), 438–441. https://doi.org/10.7916/D81V5NVX

Beckett, D., Berners-Lee, T., Prud'hommeaux, E., & Carothers, G. (2014). *RDF 1.1 Turtle* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/turtle/

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American*, *284*(5), 28–37. https://www.scientificamerican.com/article/the-semantic-web/

Bex, F. J. (2011). *Arguments, stories and criminal evidence: A formal hybrid theory* (Vol. 92). Springer. ISBN: 978-94-007-0140-3. https://doi.org/10.1007/978-94-007-0140-3

Bickmore, T., & Cassell, J. (1999). Small talk and conversational storytelling in embodied conversational interface agents. *Proceedings of the AAAI Fall Symposium on Narrative Intelligence*, 87–92. https://www.aaai.org/Papers/Symposia/Fall/1999/FS-99-01/FS99-01-015.pdf

Bizer, C., Heath, T., & Berners-Lee, T. (2011). Linked Data: The story so far. *Semantic services, interoperability and web applications: Emerging concepts* (pp. 205–227). IGI Global. https://doi.org/10.4018/978-1-60960-593-3

Björne, J., & Salakoski, T. (2011). Generalizing biomedical event extraction. *Proceedings of the BioNLP Shared Task 2011 Workshop*, 183–191. https://aclanthology.org/W11-1828.pdf

Borst, W. N. (1999). *Construction of engineering ontologies for knowledge sharing and reuse* (Doctoral dissertation). University of Twente. https://research.utwente.nl/files/6036649/t0000004.pdf

Bosma, W., Vossen, P., Soroa, A., Rigau, G., Tesconi, M., Marchetti, A., Monachini, M., & Aliprandi, C. (2009). KAF: A generic semantic annotation format. *Proceedings of the GL2009 Workshop on Semantic Annotation*, 1–8. https://adimen.si.ehu.es/~rigau/publications/gl09-kaf.pdf

Bosque-Gil, J., Gracia, J., Montiel-Ponsoda, E., & Gómez-Pérez, A. (2018). Models to represent Linguistic Linked Data. *Natural Language Engineering*, *24*(6), 811–859. https://doi.org/10.1017/S1351324918000347

Botha, J. A., Shan, Z., & Gillick, D. (2020). Entity linking in 100 languages. *arXiv:2011.02690*. https://arxiv.org/abs/2011.02690

Brickley, D., & Guha, R. (2014). *RDF schema 1.1* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/rdf-schema/

Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. *Neurocomputing* (pp. 227–236). Springer. https://doi.org/10.1007/978-3-642-76153-9_28

Bruner, J. (1991). The narrative construction of reality. *Critical inquiry*, *18*(1), 1–21.

Campos, R., Dias, G., Jorge, A. M., & Jatowt, A. (2014). Survey of temporal information retrieval and related applications. *ACM Computing Surveys (CSUR)*, *47*(2), 1–41. https://doi.org/10.1145/2619088

Cao, K. (2017). *Improving event extraction: Casting a wider net* (Doctoral dissertation). University of New York. https://cs.nyu.edu/media/publications/cao_kai.pdf

Chambers, N., & Jurafsky, D. (2008). Unsupervised learning of narrative event chains. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 789–797. https://aclanthology.org/P08-1090.pdf

Chambers, N., & Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 602–610. https://aclanthology.org/P09-1068.pdf

Chen, Y., Xu, L., Liu, K., Zeng, D., & Zhao, J. (2015). Event extraction via dynamic multi-pooling convolutional neural networks. *Proceedings of of the Annual Meeting of the Association for Computational Linguistics*, *1*, 167–176. https://aclanthology.org/P15-1017.pdf

Cheong, Y.-G., & Young, R. M. (2006). A computational model of narrative generation for suspense. *AAAI 2006*, 1906–1907. https://www.aaai.org/Papers/Workshops/2006/WS-06-04/WS06-04-003.pdf

Chiarcos, C., McCrae, J., Cimiano, P., & Fellbaum, C. (2013). Towards open data for linguistics: Linguistic Linked Data. *New trends of research in ontologies and lexical resources* (pp. 7–25). Springer. https://doi.org/10.1007/978-3-642-31782-8_2

Chiarcos, C., & Sukhareva, M. (2015). OLiA: Ontologies of linguistic annotation. *Semantic Web*, *6*(4), 379–386. https://doi.org/10.3233/SW-140167

Chieu, H. L., & Ng, H. T. (2002). A maximum entropy approach to information extraction from semi-structured and free text. *AAAI-02 Proceedings*, 786–791. https://doi.org/10.5555/777092.777213

Chinchor, N. (1998). Overview of MUC-7/MET-2. *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 – May 1, 1998*. https://aclanthology.org/M98-1001.pdf

Ciotti, F. (2018). A formal ontology for the Text Encoding Initiative. *Umanistica Digitale*, (3). https://doi.org/10.6092/issn.2532-8816/8174

Cohen, J., & Mihailidis, P. (2012). Storify and news curation: Teaching and learning about digital storytelling. *Second Annual Social Media Technology Conference & Workshop*, *1*, 27–31.

Constantin, A., Peroni, S., Pettifer, S., Shotton, D., & Vitali, F. (2016). The Document Components Ontology (DoCO). *Semantic Web*, *7*(2), 167–181. https://doi.org/10.3233/SW-150177

Corcoglioniti, F., Rospocher, M., Aprosio, A. P., & Tonelli, S. (2016). PreMOn: A Lemon extension for exposing predicate models as Linked Data. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 877–884. https://aclanthology.org/L16-1141.pdf

Crawford, C. (2012). *Chris Crawford on interactive storytelling*. New Riders. ISBN: 978-0-321-27890-6.

Cullingford, R. (1981). SAM. *Inside computer understanding: Five programs plus miniatures*. Psychology Press. ISBN: 978-0-89859-071-5.

Damiano, R., Lombardo, V., Lieto, A., & Borra, D. (2016). Exploring cultural heritage repositories with creative intelligence. the Labyrinth 3D system. *Entertainment Computing*, *16*, 41–52. https://doi.org/10.1016/j.entcom.2016.05.002

Damiano, R., Lombardo, V., & Pizzo, A. (2005). Formal encoding of drama ontology. *Proceedings of the International Conference on Virtual Storytelling*, 95–104. https://doi.org/10.1007/11590361_11

Dautenhahn, K. (2002). The origins of narrative: In search of the transactional format of narratives in humans and other social animals. *International Journal of Cognition and Technology*, *1*(1), 97–123. https://doi.org/10.1075/ijct.1.1.07dau

De Saussure, F. (1989). *Cours de linguistique générale: Édition critique* (Vol. 1). Harrassowitz Verlag. ISBN: 978-2-228-88942-1.

Delpeuch, A. (2019). OpenTapioca: Lightweight entity linking for Wikidata. *arXiv:1904.09131*. https://arxiv.org/abs/1904.09131

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*. http://arxiv.org/abs/1810.04805

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., & Weischedel, R. M. (2004). The Automatic Content Extraction (ACE) program-tasks, data, and evaluation. *Proceedings of the International Conference on Language Resources and Evaluation*, *2*, 1. http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf

Doerr, M. (2003). The CIDOC Conceptual Reference Module: An ontological approach to semantic interoperability of metadata. *AI Magazine*, *24*(3), 75–92. https://doi.org/10.1609/aimag.v24i3.1720

Doerr, M., Bekiari, C., & LeBoeuf, P. (2008). FRBRoo, a conceptual model for performing arts. *2008 Annual Conference of CIDOC*, 06–18. https://projects.ics.forth.gr/_publications/drfile.2008-06-42.pdf

Doerr, M., Gradmann, S., Hennicke, S., Isaac, A., Meghini, C., & van de Sompel, H. (2010). The Europeana Data Model (EDM). *Proceedings of the 76th World Library and Information Congress*, 10–15. https://www.ifla.org/past-wlic/2010/149-doerr-en.pdf

Doerr, M., Murano, F., & Felicetti, A. (2020). Definition of the CRMtex. http://www.cidoc-crm.org/crmtex/sites/default/files/CRMtex_v1.0_March_2020.pdf

Doerr, M., Ore, C.-E., & Stead, S. (2007). The CIDOC Conceptual Reference Model: A new standard for knowledge sharing. *Tutorials, posters, panels and industrial contributions at the 26th International Conference on Conceptual Modeling*, 51–56. http://old.cidoc-crm.org/docs/CRM_Tutorial_ER2007.pdf

Doll, W. J., & Torkzadeh, G. (1988). The measurement of end-user computing satisfaction. *MIS quarterly*, 259–274. https://doi.org/10.2307/248851

Don, A. (1990). Narrative and the interface. In B. Laurel & S. J. Mountford (Eds.), *The art of human-computer interface design* (pp. 383–391). Addison-Wesley. ISBN: 978-0-201-51797-2.

Dyer, M. G. (1982). *In-depth understanding: A computer model of integrated processing for narrative comprehension* (tech. rep.). Yale University. New Haven, Connecticut, USA. ISBN: 978-0-262-54155-8.

Eger, M., Barot, C., & Young, R. M. (2015). Impulse: A formal characterization of story. *OASIcs: OpenAccess Series in Informatics*, *45*. https://drops.dagstuhl.de/opus/volltexte/2015/5280/pdf/9.pdf

Eide, Ø. (2014). Ontologies, data modeling, and TEI. *Journal of the Text Encoding Initiative*, (8). https://doi.org/10.4000/jtei.1191

Elson, D. K. (2012). Detecting story analogies from annotations of time, action and agency. *Proceedings of the LREC Workshop on Computational Models of Narrative*. http://www.cs.columbia.edu/~delson/pubs/CMN2012-Elson.pdf

Elson, D. K., Dames, N., & McKeown, K. R. (2010). Extracting social networks from literary fiction. *Proceedings of of the Annual Meeting of the Association for Computational Linguistics*, 138–147.

Elson, D. K., & McKeown, K. R. (2007). A platform for symbolically encoding human narratives. *Intelligent Narrative Technologies*.

Erxleben, F., Günther, M., Krötzsch, M., Mendez, J., & Vrandei, D. (2014). Introducing Wikidata to the Linked Data Web. *Proceedings of ISWC*, 50–65. https://doi.org/10.1007/978-3-319-11964-9_4

Europeana. (2014). *Results of the Europeana user survey 2014* (tech. rep.). Europeana. http://pro.europeana.eu/blogpost/results-of-the-europeana-user-survey-2014

Färber, M., Bartscherer, F., Menne, C., & Rettinger, A. (2017). Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, *9*(1), 1–53. https://doi.org/10.3233/SW-170275

Farrar, S., & Langendoen, D. T. (2003). Markup and the GOLD ontology. *Workshop on Digitizing and Annotating Text and Field Recordings*. http://emeld.org/workshop/2003/langendoen-paper.pdf

Feng, X., Qin, B., & Liu, T. (2018). A language-independent neural network for event detection. *Science China Information Sciences*, *61*(9), 092106. https://aclanthology.org/P16-2011.pdf

Fernie, K., Griffiths, J., Archer, P., Chandrinos, K., de Polo, A., Stevenson, M., Clough, P., Goodale, P., Hall, M., Agirre, E., et al. (2012). PATHS: Personalising access to cultural heritage spaces. *2012 18th International Conference on Virtual Systems and Multimedia*, 469–474. https://doi.org/10.1109/VSMM.2012.6365960

Ferragina, P., & Scaiella, U. (2010). TAGME: On-the-fly annotation of short text fragments (by Wikipedia entities). *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 1625–1628. https://doi.org/10.1145/1871437.1871689

Finlayson, M. A., Richards, W., & Winston, P. H. (2010). Computational models of narrative: Review of a workshop. *AI Magazine*, *31*(2), 97. https://doi.org/10.1609/aimag.v31i2.2295

Fokkens, A., Soroa, A., Beloki, Z., Ockeloen, N., Rigau, G., Van Hage, W. R., & Vossen, P. (2014). NAF and GAF: Linking linguistic annotations. *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, 9–16.

Foppiano, L., & Romary, L. (2018). Entity-fishing: A dariah entity recognition and disambiguation service. *Proceedings of JADH*. https://doi.org/10.17928/jjadh.5.1_22

Franzosi, R. (1998). Narrative as data: Linguistic and statistical tools for the quantitative study of historical events. *International review of social history*, *43*(S6), 81–104. https://doi.org/10.1017/S002085900011510X

Freitag, D. (1998). Multistrategy learning for information extraction. *Proceedings of the International Conference on Machine Learning*, 161–169.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., & Schneider, L. (2002). Sweetening ontologies with DOLCE. *Knowledge engineering and knowledge management: Ontologies and the Semantic Web*, 223–233. https://doi.org/10.1007/3-540-45810-7_18

Garbacz, P. (2016). A formal ontology of texts. *Proceedings of the International Conference on Formal Ontology in Information Systems*, 345–358. https://doi.org/10.3233/978-1-61499-660-6-345

Genette, G. E., & Lewin, J. E. (1983). *Narrative discourse: An essay in method*. Cornell University Press. ISBN: 978-0-8014-9259-4.

Greimas, A. J., McDowell, D., & Velie, A. R. (1983). *Structural semantics: An attempt at a method*. University of Nebraska Press. ISBN: 978-0-8032-2112-3.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, *5*(2), 199–220. https://doi.org/10.1006/knac.1993.1008

Guarino, N., Oberle, D., & Staab, S. (2009). What is an ontology? *Handbook on ontologies* (pp. 1–17). Springer. https://doi.org/10.1007/978-3-540-92673-3_0

Hausner, P., Aumiller, D., & Gertz, M. (2020). Time-centric exploration of court documents. *Text2Story at ECIR*, 31–37. http://ceur-ws.org/Vol-2593/paper4.pdf

Hellmann, S., Lehmann, J., Auer, S., & Brümmer, M. (2013). Integrating NLP using Linked Data. *International Semantic Web Conference*, 98–113. https://doi.org/10.1007/978-3-642-41338-4_7

Herman, D. (2000). Narratology as a cognitive science. *Image and Narrative*, *1*, 1. http://www.imageandnarrative.be/inarchive/narratology/davidherman.htm

Herman, D. (2009). *Basic elements of narrative*. Wiley-Blackwell, Hoboken, NJ. ISBN: 978-1-4443-0593-7.

Hirsch, C., Hosking, J., & Grundy, J. (2009). Interactive visualization tools for exploring the semantic graph of large knowledge spaces. *Workshop on Visual Interfaces to the Social and the Semantic Web*, *443*. http://ceur-ws.org/Vol-443/paper6.pdf

Hobbs, J. R., & Pan, F. (2020). *Time ontology in owl* (S. Cox & C. Little, Eds.; W3C Candidate Recommendation). World Wide Web Consortium. https://www.w3.org/TR/owl-time

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Horridge, M., & Bechhofer, S. (2011). The OWL API: A Java api for OWL ontologies. *Semantic Web*, *2*(1), 11–21. https://doi.org/10.3233/SW-2011-0025

Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., & Dean, M. (2004). *SWRL: A Semantic Web rule language combining OWL and RuleML* (W3C Member Submission). World Wide Web Consortium. https://www.w3.org/Submission/SWRL

Hyvärinen, M. (2006). Towards a conceptual history of narrative. *COLLeGIUM*. https://helda.helsinki.fi/bitstream/handle/10138/25742/001_04_hyvarinen.pdf

Hyvönen, E., Leskinen, P., Tamper, M., Rantala, H., Ikkala, E., Tuominen, J., & Keravuori, K. (2019). Biographysampo: Publishing and enriching biographies on the Semantic Web for Digital Humanities research. *European Semantic Web Conference*, 574–589. https://doi.org/10.1007/978-3-030-21348-0_37

Hyvönen, E., Takala, J., Alm, O., Ruotsalo, T., & Mäkelä, E. (2007). Semantic Kalevala: Accessing cultural contents through semantically annotated stories. *Proceedings of the Cultural Heritage on the Semantic Web Workshop at the 6th International Semantic Web Conference (ISWC 2007), Busan, Korea, Nov, 2007*. https://seco.cs.aalto.fi/publications/2007/hyvonen-et-al-kalevala-2007.pdf

Kolodner, J. L. (1980). *Retrieval and organizational strategies in conceptual memory: A computer model* (tech. rep.). Yale University. New Haven, Connecticut, USA. ISBN: 978-1-138-99734-9.

Komninos, A., & Manandhar, S. (2016). Dependency based embeddings for sentence classification tasks. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1490–1500. https://aclanthology.org/N16-1175.pdf

Kreiswirth, M. (2000). Merely telling stories? narrative and knowledge in the human sciences. *Poetics Today*, *21*(2), 293–318. https://doi.org/10.1215/03335372-21-2-293

Labov, W., & Waletzky, J. (1997). Narrative analysis: Oral versions of personal experience. *Journal of Narrative & Life History*. https://doi.org/10.1075/jnlh.7.02nar

Lakoff, G., & Narayanan, S. (2010). Toward a computational model of narrative. *Proceedings of the AAAI Fall Symposium on Computational Models of Narrative*, 21–28. https://www.aaai.org/ocs/index.php/FSS/FSS10/paper/view/2323

Laurel, B. (1991). *Computers as theatre*. Addison-Wesley. ISBN: 978-0-201-55060-3.

Lawrence, D., & Thomas, J. (1999). Social dynamics of storytelling: Implications for story-base design. *Proceedings of the AAAI Workshop on Narrative Intelligence*. https://www.cs.cmu.edu/~michaelm/nidocs/LawrenceThomas.pdf

Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., & Zhao, J. (2013). PROV-O: The prov ontology. https://www.w3.org/TR/prov-o

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436. https://doi.org/10.1038/nature14539

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., & Bizer, C. (2015). DBpedia: A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, *6*(2), 167–195. https://doi.org/10.3233/SW-140134

Lehnert, W., Cardie, C., Fisher, D., McCarthy, J., Riloff, E., & Soderland, S. (1992). University of Massachusetts: MUC-4 test results and analysis. *Proceedings of the Message Understanding Conference*, 151–158. https://aclanthology.org/M92-1021.pdf

Levesque, H. J., & Lakemeyer, G. (2001). *The logic of knowledge bases*. The MIT Press, Cambridge, MA. ISBN: 978-0-262-12232-0.

Levi-Strauss, C. (1963). Structural analysis in linguistics and in anthropology. *Semiotics: An Introductory Anthology*, 110–128. ISBN: 978-1-315-08227-1.

Levitin, A. (2007). *Introduction to the design & analysis of algorithms*. Pearson Education, Addison-Wesley, Upper Saddle River, NJ. ISBN: 978-0-13-231681-1.

Li, Q., Ji, H., & Huang, L. (2013). Joint event extraction via structured prediction with global features. *Proceedings of of the Annual Meeting of the Association for Computational Linguistics*, *1*, 73–82. https://aclanthology.org/P13-1008.pdf

Linger, M., & Hajaiej, M. (2020). Batch clustering for multilingual news streaming. *arXiv:2004.08123*. https://arxiv.org/abs/2004.08123

Liu, J., Chen, Y., Liu, K., Bi, W., & Liu, X. (2020). Event extraction as machine reading comprehension. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1641–1651. https://aclanthology.org/2020.emnlp-main.128.pdf

Lloyd, J. W. (1984). *Foundations of logic programming*. Springer-Verlag. ISBN: 978-3-642-96826-6. https://doi.org/10.1007/978-3-642-96826-6

Lombardo, V., & Pizzo, A. (2014). Ontology–based visualization of characters' intentions. *Proceedings of the International Conference on Interactive Digital Storytelling*, 176–187. https://doi.org/10.1007/978-3-319-12337-0_18

Magerko, B. S., & Riedl, M. O. (2007). *Intelligent narrative technologies: Papers from the aaai fall symposium* (tech. rep.). Arlington, Virginia, USA, AAAI. https://www.aaai.org/Press/Reports/Symposia/Fall/fs-07-05.php

Mani, I. (2012). Computational modeling of narrative. *Synthesis Lectures on Human Language Technologies*, *5*(3), 1–142. https://doi.org/10.2200/S00459ED1V01Y201212HLT018

Mani, I. (2014). Computational narratology. In P. e. a. Hühn (Ed.), *The living handbook of narratology*. Hamburg University. http://www.lhn.uni-hamburg.de/article/computational-narratology

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. *Proceedings of of the Annual Meeting of the Association for Computational Linguistics*, 55–60. https://aclanthology.org/P14-5010.pdf

Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press. ISBN: 978-0-262-13360-9.

Mannocci, A., Casarosa, V., Manghi, P., & Zoppi, F. (2014). The Europeana network of ancient Greek and Latin epigraphy data infrastructure. *Research Conference on Metadata and Semantics Research*, 286–300.

Manola, F., & Miller, E. (2004). *RDF Primer* (W3C Recommendation). World Wide Web Consortium. http://www.w3.org/TR/rdf-primer

Mateas, M., & Sengers, P. (2003). *Narrative intelligence*. John Benjamins. https://doi.org/10.1075/aicr.46

McCrae, J., Spohr, D., & Cimiano, P. (2011). Linking lexical resources and ontologies on the Semantic Web with Lemon. *Extended Semantic Web Conference*, 245–259. https://doi.org/10.1007/978-3-642-21034-1_17

McCrae, J. P., Bosque-Gil, J., Gracia, J., Buitelaar, P., & Cimiano, P. (2017). The Ontolex-Lemon model: Development and applications. *Proceedings of eLex 2017 conference*, 19–21. http://john.mccr.ae/papers/mccrae2017ontolex.pdf

McGuinness, D. L., & Van Harmelen, F. (2004). OWL Web Ontology Language overview. https://www.w3.org/TR/owl-features

Meehan, J. R. (1976). *The metanovel: Writing stories by computer* (Doctoral dissertation). Yale University. New Haven, Connecticut, USA.

Meghini, C. (2020). *The Mingei approach: We collect, connect and open up* (tech. rep.). Mingei Newsletter. http://www.mingei-project.eu/the-mingei-approach

Meghini, C., & Doerr, M. (2018). A first-order logic expression of the CIDOC Conceptual Reference Model. *International Journal of Metadata, Semantics and Ontologies*, *13*(2), 131–149. https://doi.org/10.1504/IJMSO.2018.098393

Meghini, C., Bartalesi, V., & Metilli, D. (2021). Representing narratives in digital libraries: The Narrative Ontology. *Semantic Web*, *12*(2), 241–264. https://doi.org/10.3233/SW-200421

Meghini, C., Bartalesi, V., Metilli, D., & Benedetti, F. (2017). Introducing narratives in Europeana: Preliminary steps. *Advances in Databases and Information Systems*, 333–342. https://doi.org/10.1007/978-3-319-67162-8_33

Meghini, C., Bartalesi, V., Metilli, D., & Benedetti, F. (2019). Introducing narratives in Europeana: A case study. *International Journal of Applied Mathematics and Computer Science*, *29*(1), 7–16. https://doi.org/10.2478/amcs-2019-0001

Meister, J. C. (2003). *Computing action: A narratological approach* (Vol. 2). Walter de Gruyter. ISBN: 978-3-11-017628-5.

Meister, J. C. (2014). Narratology. In P. Hühn, J. C. Meister, J. Pier, & W. Schmid (Eds.), *The living handbook of narratology*. University of Hamburg, Hamburg. http://www.lhn.uni-hamburg.de/article/narratology

Mellace, S., Antonucci, A. et al. (2020). Temporal embeddings and transformer models for narrative text understanding. *arXiv:2003.08811*. https://arxiv.org/abs/2003.08811

Metilli, D. (2016). *A Wikidata-based tool for the creation of narratives* (Master's thesis). University of Pisa. https://etd.adm.unipi.it/theses/available/etd-09052016-094429/

Metilli, D., Bartalesi, V., & Meghini, C. (2019a). Steps towards a system to extract formal narratives from text. *Text2Story at ECIR*, 53–61. http://ceur-ws.org/Vol-2342/paper7.pdf

Metilli, D., Bartalesi, V., & Meghini, C. (2019b). A Wikidata-based tool for building and visualising narratives. *International Journal on Digital Libraries*, *20*(4), 417–432. https://doi.org/10.1007/s00799-019-00266-3

Metilli, D., Bartalesi, V., Meghini, C., & Aloia, N. (2019). Populating narratives using Wikidata events: An initial experiment. *Digital Libraries: Supporting Open Science – 15th Italian Research Conference on Digital Libraries, IRCDL 2019, Pisa, Italy, January 31 – February 1, 2019, Proceedings*. https://doi.org/10.1007/978-3-030-11226-4

Mietchen, D., Hagedorn, G., Willighagen, E., Rico, M., Gómez-Pérez, A., Aibar, E., Rafes, K., Germain, C., Dunning, A., Pintscher, L., & Kinzler, D. (2015). Enabling open science: Wikidata for Research (Wiki4R). *Research Ideas and Outcomes*, *1*. https://doi.org/10.3897/RIO.1.E7573

Minard, A.-L., Speranza, M., Urizar, R., Altuna, B., van Erp, M., Schoen, A., van Son, C., et al. (2016). MEANTIME, the NewsReader multilingual event and time corpus. *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 4417. https://aclanthology.org/L16-1699.pdf

Möller, C., Lehmann, J., & Usbeck, R. (n.d.). *Survey on English entity linking on Wikidata*. http://www.semantic-web-journal.net/system/files/swj2865.pdf

Moorman, K., & Ram, A. (1993). A new perspective on story understanding. *Proceedings of the 31st Southeast ACM Conference*. https://www.cc.gatech.edu/faculty/ashwin/papers/er-93-02.pdf

Motik, B., Patel-Schneider, P. F., & Parsia, B. (2012). *OWL 2 Web Ontology Language structural specification and functional-style syntax (second edition)* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/owl-syntax

Mulholland, P., & Collins, T. (2002). Using digital narratives to support the collaborative learning and exploration of cultural heritage. *13th International Workshop on Database and Expert Systems Applications*, 527–531. https://doi.org/10.1109/DEXA.2002.1045951

Murray, J. H. (2017). *Hamlet on the holodeck: The future of narrative in cyberspace*. MIT Press. ISBN: 978-0-262-53348-5.

Nguyen, K.-H., Tannier, X., Ferret, O., & Besançon, R. (2016). A dataset for open event extraction in English. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 1939–1943. https://aclanthology.org/L16-1307.pdf

Nguyen, T. H., Cho, K., & Grishman, R. (2016). Joint event extraction via recurrent neural networks. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 300–309. https://aclanthology.org/N16-1034.pdf

Nguyen, T. H., & Grishman, R. (2015). Event detection and domain adaptation with convolutional neural networks. *Proceedings of of the Annual Meeting of the Association for Computational Linguistics*, *2*, 365–371. https://aclanthology.org/P15-2060.pdf

Niehaus, J., Romero, V., Koelle, D., Palmon, N., Bracken, B., Pfautz, J., Reilly, S. N., & Weyhrauch, P. (2014). A flexible framework for the creation of narrative-centered tools. *OASIcs: OpenAccess Series in Informatics*, *41*. https://drops.dagstuhl.de/opus/volltexte/2014/4651/pdf/18.pdf

Oldman, D., & Tanase, D. (2018). Reshaping the knowledge graph by connecting researchers, data and practices in ResearchSpace. *The Semantic Web – ISWC 2018*, 325–340. https://doi.org/10.1007/978-3-030-00668-6_20

Ore, C.-E., & Eide, Ø. (2009). TEI and cultural heritage ontologies: Exchange of information? *Literary and Linguistic Computing*, *24*(2), 161–172. https://doi.org/10.1093/llc/fqp010

Partarakis, N., Doulgeraki, P., Karuzaki, E., Adami, I., Ntoa, S., Metilli, D., Bartalesi, V., Meghini, C., Marketakis, Y., Theodoridou, M., D., K., & Zabulis, X. (n.d.). Representation of socio-historical context to support the authoring and presentation of multimodal narratives: The Mingei Online Platform [Accepted for publication]. *Journal on Computing and Cultural Heritage*.

Partarakis, N., Kaplanidi, D., Doulgeraki, P., Karuzaki, E., Petraki, A., Metilli, D., Bartalesi, V., Adami, I., Meghini, C., & Zabulis, X. (2021). Representation and presentation of culinary tradition as cultural heritage. *Heritage*, *4*(2), 612–640. https://doi.org/10.3390/heritage4020036

Partarakis, N., Patsiouras, N., Evdemon, T., Doulgeraki, P., Karuzaki, E., Stefanidi, E., Ntoa, S., Meghini, C., Kaplanidi, D., Fasoula, M., et al. (2020). Enhancing the educational value of tangible and intangible dimensions of traditional crafts through role-play gaming. *International Conference on ArtsIT, Interactivity and Game Creation*, 243–254. https://doi.org/10.1007/978-3-030-73426-8_14

Parvizi, A., Kohl, M., Gonzàlez, M., & Saur, R. (2016). Towards a linguistic ontology with an emphasis on reasoning and knowledge reuse. *Proceedings of the Tenth*

International Conference on Language Resources and Evaluation (LREC'16), 441–448. https://aclanthology.org/L16-1071.pdf

Peroni, S., & Shotton, D. (2012). FaBiO and CiTO: Ontologies for describing bibliographic resources and citations. *Journal of Web Semantics*, *17*, 33–43. https://doi.org/10.1016/j.websem.2012.08.001

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv:1802.05365*. https://arxiv.org/abs/1802.05365

Piccinno, F., & Ferragina, P. (2014). From TagME to WAT: A new entity annotator. *Proceedings of the First International Workshop on Entity Recognition & Disambiguation*, 55–62. https://doi.org/10.1145/2633211.2634350

Piscopo, A., Phethean, C., & Simperl, E. (2017). Wikidatians are born: Paths to full participation in a collaborative structured knowledge base. *Proceedings of the 50th Hawaii International Conference on System Sciences*. https://doi.org/10.24251/HICSS.2017.527

Piskorski, J., Zavarella, V., Atkinson, M., & Verile, M. (2020). Timelines: Entity-centric event extraction from online news. *Text2Story at ECIR*, 105–114. http://ceur-ws.org/Vol-2593/paper13.pdf

Propp, V. (2010). *Morphology of the folktale*. University of Texas Press. ISBN: 978-0-292-78376-8.

Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., Katz, G., & Radev, D. R. (2003). TimeML: Robust specification of event and temporal expressions in text. *New directions in question answering*, *3*, 28–34. https://www.aaai.org/Papers/Symposia/Spring/2003/SS-03-07/SS03-07-005.pdf

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A python natural language processing toolkit for many human languages. *arXiv:2003.07082*. https://arxiv.org/abs/2003.07082

Raimond, Y., & Abdallah, S. (2007). *The Event Ontology* (tech. rep.). Queen Mary University of London. http://motools.sourceforge.net/event

Ram, A. (1989). *Question-driven understanding: An integrated theory of story understanding, memory and learning* (Doctoral dissertation). Yale University. https://cpsc.yale.edu/sites/default/files/files/tr710.pdf

Reimers, N., & Gurevych, I. (2017). Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. *arXiv:1707.09861*. https://arxiv.org/abs/1707.09861

Reimers, N., & Gurevych, I. (2019). Alternative Weighting Schemes for ELMo Embeddings. *arXiv:1904.02954*. https://arxiv.org/abs/1904.02954

Renz, J., & Nebel, B. (2007). Qualitative spatial reasoning using constraint calculi. *Handbook of spatial logics* (pp. 161–215). Springer. https://doi.org/10.1007/978-1-4020-5587-4_4

Rishes, E., Lukin, S. M., Elson, D. K., & Walker, M. A. (2013). Generating different story tellings from semantic representations of narrative. *Proceedings of the International Conference on Interactive Digital Storytelling*, 192–204. https://doi.org/10.1007/978-3-319-02756-2_24

Riva, P., Doerr, M., & Zumer, M. (2008). FRBRoo: Enabling a common view of information from memory institutions. *Proceedings of the IFLA World Library and Information Congress*. https://archive.ifla.org/IV/ifla74/papers/156-Riva_Doerr_Zumer-en.pdf

Riva, P., & Zumer, M. (2017). FRBRoo, the IFLA Library Reference Model, and now LRMoo: A circle of development. http://library.ifla.org/id/eprint/2130/1/074-riva-en.pdf

Rubiera, E., Polo, L., Berrueta, D., & El Ghali, A. (2012). TELIX: An RDF-based model for linguistic annotation. *Extended Semantic Web Conference*, 195–209. https://doi.org/10.1007/978-3-642-30284-8_20

Rumelhart, D. E. (1975). Notes on a schema for stories. *Representation and understanding: Five programs plus miniatures* (pp. 211–236). Elsevier. https://doi.org/10.1016/B978-0-12-108550-6.50013-6

Russell, S. J., Norvig, P., Canny, J. F., Malik, J. M., & Edwards, D. D. (2003). *Artificial intelligence: A modern approach*. Prentice Hall. ISBN: 978-0-13-461099-3.

Ryan, M.-L. (1991). *Possible worlds, artificial intelligence, and narrative theory*. Indiana University Press. ISBN: 978-0-253-35004-6.

Ryan, M.-L. (2007). Toward a definition of narrative. *The Cambridge companion to narrative*, 22–35. https://doi.org/10.1017/CCOL0521856965.002

Sack, W. (2002). Stories and social networks. *Advances in Consciousness Research*, *46*, 305–322. https://doi.org/10.1075/aicr.46.22sac

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake shakes Twitter users: Real-time event detection by social sensors. *Proceedings of the 19th International Conference on World Wide Web*, 851–860. https://doi.org/10.1145/1772690.1772777

Sakor, A., Singh, K., Patel, A., & Vidal, M.-E. (2020). Falcon 2.0: An entity and relation linking tool over Wikidata. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 3141–3148. https://doi.org/10.1145/3340531.3412777

Salehpour, M., & Davis, J. G. (2020). A comparative analysis of knowledge graph query performance. *arXiv:2004.05648*. https://arxiv.org/abs/2004.05648

Sanderson, R., Ciccarese, P., Van de Sompel, H., Bradshaw, S., Brickley, D., a Castro, L. J. G., Clark, T., Cole, T., Desenne, P., Gerber, A., et al. (2013). Open Annotation Data Model. http://www.openannotation.org/spec/core

Sanderson, R., Ciccarese, P., & Young, B. (2017). Web Annotation Data Model. https://www.w3.org/TR/annotation-model

Schank, R. C. (1990). *Tell me a story: A new look at real and artificial memory*. Scribner. ISBN: 978-0-684-19049-5.

Schank, R. C., & Abelson, R. P. (1975). Scripts, plans, and knowledge. *Proceedings of the International Joint Conference on Artificial Intelligence*, 151–157. https://home.mis.u-picardie.fr/~furst/docs/Schank_Abelson_Scripts_1975.pdf

Scherp, A., Franz, T., Saathoff, C., & Staab, S. (2009). F: A model of events based on the foundational ontology Dolce+ DnS ultralight. *Proceedings of the fifth International Conference on Knowledge Capture*, 137–144. https://doi.org/10.1145/1597735.1597760

Schmitt, X., Kubler, S., Robert, J., Papadakis, M., & LeTraon, Y. (2019). A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, 338–343. https://doi.org/10.1109/SNAMS.2019.8931850

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093

Sengers, P. (1998). *Anti-boxology: Agent design in cultural context* (Doctoral dissertation). Carnegie-Mellon University. https://apps.dtic.mil/sti/pdfs/ADA363523.pdf

Shaw, R., Troncy, R., & Hardman, L. (2009). LODE: Linking Open Descriptions of Events. *The Semantic Web* (pp. 153–167). Springer. ISBN: 978-3-642-10871-6. https://doi.org/10.1007/978-3-642-10871-6_11

Shekarpour, S., Alshargi, F., Thirunaravan, K., Shalin, V. L., & Sheth, A. (2019). CEVO: Comprehensive event ontology enhancing cognitive annotation on relations. *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, 385–391. https://doi.org/10.1109/ICOSC.2019.8665605

Shekarpour, S., Saxena, A., Thirunarayan, K., Shalin, V. L., & Sheth, A. (2018). Principles for developing a knowledge graph of interlinked events from news headlines on Twitter. *arXiv:1808.02022*. https://arxiv.org/abs/1808.02022

Shen, W., Wang, J., & Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 443–460. https://doi.org/10.1109/TKDE.2014.2327028

Shklovsky, V. (1997). Art as technique. In K. Newton (Ed.), *Twentieth-century literary theory* (pp. 3–5). Palgrave, London. ISBN: 978-1-349-25934-2. https://doi.org/10.1007/978-1-349-25934-2_1

Shotton, D. (2010). CiTO, the citation typing ontology. *Journal of biomedical semantics*, *1*(1), 1–18. https://doi.org/10.1186/2041-1480-1-S1-S6

Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., & Katz, Y. (2007). Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics*, *5*(2), 51–53. https://doi.org/10.1016/j.websem.2007.03.004

Smullyan, R. R. (1968). *First-order logic* (Vol. 43). Springer. ISBN: 978-3-642-86718-7. https://doi.org/10.1007/978-3-642-86718-7

Spierling, U., & Szilas, N. (2008). *Interactive storytelling: First Joint International Conference on Interactive Digital Storytelling*. Springer. https://doi.org/10.1007/978-3-540-89454-4

Strötgen, J., & Gertz, M. (2010). Heideltime: High quality rule-based extraction and normalization of temporal expressions. *Proceedings of the 5th International Workshop on Semantic Evaluation*, 321–324. https://aclanthology.org/S10-1071.pdf

Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & knowledge engineering*, *25*(1-2), 161–197. https://doi.org/10.1016/S0169-023X(97)00056-6

Suchanek, F. M., Kasneci, G., & Weikum, G. (2007). YAGO: A core of semantic knowledge. *Proceedings of the 16th International Conference on World Wide Web*, 697–706. https://doi.org/10.1145/1242572.1242667

Szilas, N. (1999). Interactive drama on computer: Beyond linear narrative. *Proceedings of the AAAI Fall Symposium on Narrative Intelligence*, *144*, 150–156. https://aaai.org/Papers/Symposia/Fall/1999/FS-99-01/FS99-01-026.pdf

Szilas, N. (2007). A computational model of an intelligent narrator for interactive narratives. *Applied Artificial Intelligence*, *21*(8), 753–801. https://doi.org/10.1080/08839510701526574

Tavoni, M. (2012). Dantesearch: Il corpus delle opere volgari e latine di dante lemmatizzate con marcatura grammaticale e sintattica. *Lectura dantis 2002–2009*. Università degli Studi di Napoli "L'Orientale". ISBN: 978-88-95044-90-3.

Taylor, C. (1989). *Sources of the self*. Harvard University Press. ISBN: 978-0-674-82426-3.

TEI Consortium (Ed.). (2021). *TEI P5: Guidelines for electronic text encoding and interchange* (tech. rep.). TEI Consortium. https://tei-c.org/guidelines/P5/

Todorov, T. (1969). *Grammaire du Décaméron*. Mouton. ISBN: 978-3-11-098134-6.

Tomazzoli, G. (2019). «Totus poema eius ubique mirabiliter figuratus». identifying, classifying and describing Dante's metaphors. *Umanistica Digitale*, (6). https://doi.org/10.6092/issn.2532-8816/8630

Tomazzoli, G., & Meghini, C. (n.d.). Per un'ontologia delle metafore dantesche [Accepted for publication]. In L. Livraghi & G. Tomazzoli (Eds.), *«Per intelletto umano e per autoritadi». Il contesto di formazione e diffusione culturale del poema dantesco*. Franco Cesati.

Tuominen, J. A., Hyvönen, E. A., Leskinen, P., et al. (2018). Bio crm: A data model for representing biographical data for prosopographical research. *Proceedings of the Second Conference on Biographical Data in a Digital World 2017 (BD2017)*. http://ceur-ws.org/Vol-2119/paper10.pdf

Tuominen, J. A., Mäkelä, E., Hyvönen, E. A., Bosse, A., Lewis, M., Hotson, H., et al. (2018). Reassembling the Republic of Letters: A Linked Data approach. *Proceedings of the Digital Humanities in the Nordic Countries 3rd Conference (DHN 2018)*. http://ceur-ws.org/Vol-2084/paper6.pdf

Turkle, S. (1991). *The second self: Computers and the human spirit*. Simon & Schuster. ISBN: 978-0-671-46848-4.

Turner, S. R. (1993). *MINSTREL: A computer model of creativity and storytelling* (Doctoral dissertation). University of California at Los Angeles.

Van Hage, W. R., Malaisé, V., Segers, R., Hollink, L., & Schreiber, G. (2011). Design and use of the Simple Event Model (SEM). *Web Semantics: Science, Services and Agents on the World Wide Web*, *9*(2), 128–136. https://doi.org/10.1016/j.websem.2011.03.003

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010. https://doi.org/10.5555/3295222.3295349

Vossen, P., Caselli, T., & Kontzopoulou, Y. (2015). Storylines for structuring massive streams of news. *Proceedings of the First Workshop on Computing News Storylines*, 40–49. https://doi.org/10.18653/v1/W15-4507

Vossen, P., Rigau, G., Serafini, L., Stouten, P., Irving, F., & Van Hage, W. R. (2014). Newsreader: Recording history from daily news streams. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 2000–2007. http://www.lrec-conf.org/proceedings/lrec2014/pdf/436_Paper.pdf

Vrandei, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, *57*(10), 78–85. https://doi.org/10.1145/2629489

W3C OWL Working Group (Ed.). (2012). *OWL 2 Web Ontology Language document overview (second edition)* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/owl2-overview

W3C MultilingualWeb-LT Working Group (Ed.). (2013). *Internationalization Tag Set (ITS) version 2.0* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/its20

W3C SPARQL Working Group (Ed.). (2013). *SPARQL 1.1 overview* (W3C Recommendation). World Wide Web Consortium. https://www.w3.org/TR/sparql11-overview

Wadden, D., Wennberg, U., Luan, Y., & Hajishirzi, H. (2019). Entity, relation, and event extraction with contextualized span representations. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5788–5793. https://doi.org/10.18653/v1/D19-1585

Wadsworth, B. J. (1996). *Piaget's theory of cognitive and affective development: Foundations of constructivism*. Longman Publishing. ISBN: 978-0-205-40603-6.

Wagner, E. J. (2010). A preliminary analysis and catalog of thematic labels. *Proceedings of the AAAI Fall Symposium on Computational Models of Narrative*. https://www.aaai.org/ocs/index.php/FSS/FSS10/paper/view/2300

Walker, C., & Medero, J. (2005). *ACE 2005 multilingual training data v6.0 LDC2005E18* (tech. rep.). Linguistic Data Consortium. Philadelphia. https://doi.org/10.35111/mwxc-vh88

Wang, S., & Zhan, H. (2010). Enhancing teaching and learning with digital storytelling. *International Journal of Information and Communication Technology Education (IJICTE)*, *6*(2), 76–87. https://doi.org/10.4018/jicte.2010040107

Wang, X. D., Weber, L., & Leser, U. (2020). Biomedical event extraction as multi-turn question answering. *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*, 88–96. https://aclanthology.org/2020.louhi-1.10.pdf

Weibel, S., Kunze, J., Lagoze, C., & Wolf, M. (1998). Dublin Core metadata for resource discovery. *Internet Engineering Task Force RFC*, *2413*(222), 132. https://doi.org/10.17487/RFC2413

Weng, J., & Lee, B.-S. (2011). Event detection in Twitter. *Proceedings of ICWSM*, *11*, 401–408. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2767/3299

Wilensky, R. (1981). PAM. *Inside computer understanding: Five programs plus miniatures*. Psychology Press. ISBN: 978-0-89859-071-5.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). HuggingFace's transformers: State-of-the-art natural language processing. *arXiv:1910.03771*. https://arxiv.org/abs/1910.03771

Wolff, A., Mulholland, P., & Collins, T. (2012). Storyspace: A story-driven approach for creating museum narratives. *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, 89–98. https://doi.org/10.1145/2309996.2310012

Yakushiji, A., Tateisi, Y., Miyao, Y., & Tsujii, J.-i. (2000). Event extraction from biomedical papers using a full parser. *Biocomputing 2001* (pp. 408–419). World Scientific. ISBN: 978-981-4447-36-2. https://doi.org/10.1142/9789814447362_0040

Yang, S., Feng, D., Qiao, L., Kan, Z., & Li, D. (2019). Exploring pre-trained language models for event extraction and generation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5284–5294. https://aclanthology.org/P19-1522.pdf

Zabulis, X., Meghini, C., Partarakis, N., Beisswenger, C., Dubois, A., Fasoula, M., Nitti, V., Ntoa, S., Adami, I., Chatziantoniou, A., et al. (2020). Representation and preservation of heritage crafts. *Sustainability*, *12*(4), 1461. https://doi.org/10.3390/su12041461

Zabulis, X., Meghini, C., Partarakis, N., Kaplanidi, D., Doulgeraki, P., Karuzaki, E., Stefanidi, E., Evdemon, T., Metilli, D., Bartalesi, V., et al. (2019). What is needed to digitise knowledge on heritage crafts? *Memoriamedia Review*. https://doi.org/10.5281/zenodo.3723792

Zarri, G. P. (2009). *Representation and management of narrative information: Theoretical principles and implementation*. Springer. ISBN: 978-1-84800-078-0. https://doi.org/10.1007/978-1-84800-078-0