Departamento de Inteligencia Artificial
Escuela Técnica Superior de Ingenieros Informáticos

PhD Thesis

Processing, Identification and Representation of Temporal Expressions and Events in Legal Documents

Author: María Navas Loro

Supervisors: Dr. Víctor Rodríguez Doncel
Dra. Asunción Gómez Pérez

October, 2021

Tribunal nombrado por el Sr. Rector Magfco. de la Universidad Politécnica de Madrid, el día DD de MM de YYYY

Presidente:

Vocal:

Vocal:

Vocal:

Secretario:

Suplente:

Suplente:

Realizado el acto de defensa y lectura de la Tesis el día XX de MM de YYYY en la Escuela Técnica Superior de Ingenieros Informáticos

Calificación: ______________________________

EL PRESIDENTE VOCAL 1 VOCAL 2

VOCAL 3 EL SECRETARIO

A mis padres y a mi hermano.
A David.

Agradecimientos

Acabar una fase vital siempre es una mezcla de alivio e incertidumbre. Cuando acabé el máster, no sabía si lanzarme a la industria o realizar una tesis doctoral; de hecho, pregunté a Óscar si me podía recomendar una empresa de NLP y, días después, me avisó de que quedaba libre una vacante en el grupo, por si me interesaba hacer la tesis en el OEG. Fue lo que se suele llamar un "perfect timing" (sobre todo teniendo en cuenta la temática de esta tesis), y éste es el resultado.

En primer lugar, me gustaría dar las gracias a Óscar y Asun, al primero por darme la oportunidad de unirme al grupo, y a la segunda por directamente convertir la entrevista para entrar en el grupo en una planificación del primer año de doctorado, además de ser también mi directora de tesis junto con Víctor. No obstante, mi mayor agradecimiento es sin duda para Víctor, con quien no tengo reuniones, sino conversaciones, y que no es sólo director, sino también compañero. Me ha ayudado a recorrer el enrevesado sendero de la academia, convirtiendo en dibujos con bolis de colorines los vericuetos de la ciencia y la burocracia, aderezando todo esto con etimología y un poco de Wittgenstein. Aunque en los informes anuales siempre reporte que soy muy independiente, sin duda gran parte del mérito de esta tesis es suyo.

Por supuesto, también quiero que darle las gracias a postdocs como Elena, que hasta como IP sigue siendo amable e intenta que las tareas más arduas sean llevaderas, y a aquellos con los que apenas compartía tema o tiempo, pero que siempre estaban dispuestos a ayudar y a mandarte cualquier cosa que pensaran que podía serte de utilidad... demasiados nombres para tan pocas hojas. Una mención especial en el laboratorio es, por supuesto, para quienes ya superaron la transición doctorando-doctor, que siempre han ayudado anímica y burocráticamente a los que seguimos sus pasos, destacando a Pablo y Julia, y a los compañeros de trinchera en los constantes lances contra el papeleo, como Patri, Carlos, Paola, Alba, o Elvira. Por último, pero no menos importante, también unas palabras para JARG, Ana y Raúl, que han visto pasar generaciones y generaciones de doctorandos y siempre tienen buenos consejos que dar desde las entrañas del laboratorio en las que soportan discretamente el entramado del grupo.

Also a few words for the friends that the academy brought me outside the group, both in projects and stays; to Erwin, with whom I already count on every visit to Vienna; to Cristiana, who can make a 5-hour call on legal judgments fun; to all the people who, in short, helped and guided me during the stays and tasks, and who again do not fit in a few lines: Sabrina, Satoh-sensei, Tiago, the people of Lynx, Bologna and Roma. I hope to be able to collaborate with all of you again in the future, and that our paths will cross again in the uncertain and ubiquitous Science. Finally, a big thank you to the reviewers of the dissertation, Natalya and Julián, who kindly invested their time to get the best version of this thesis.

Cambiando de tercio, ya que no todo es ciencia en esta vida, a mis amigos, que son pocos, dispersos e irremplazables. A pesar de su dudoso gusto en lo que a compañía y humor se refiere (alguno incluso intentó leerse la tesis), me han acompañado en las distintas disquisiciones y entelequias de la vida.

Por último, y muy especialmente, gracias a David, que me ha apoyado cada año de la tesis y más allá, y a mi familia, a la que está y a la que ya no está. Sobre todo, gracias a mis padres por enseñarme a trabajar por lo que se quiere. Nunca me regalaron un perro, pero me dieron todo lo demás.

Abstract

Legal documents can be long, complex and difficult to understand. However, there is a strong demand for access to legal information, and thousands of documents are published every day. Although there is a multitude of institutional portals available to citizens and legal practitioners, the documents themselves are often plain texts from whom it is difficult to extract information. The retrieval of temporal information in judgments is particularly important, and the analysis of these texts often requires identifying dates and events. In fact, being able to represent a sentence as a set of relevant events would be extremely useful, as it would improve searches and facilitate the visualization and understanding of texts through summaries and timelines, among others. However, there is currently no system that facilitates the processing of temporal information in legal documents.

This doctoral thesis aims to provide a framework that addresses the problem comprehensively, proposing algorithms for the recognition of temporal expressions and events, describing a data model for their representation and demonstrating that they facilitate the retrieval of temporal information in legal texts.

The main contributions are (1) several annotated corpora in the legal domain, (2) a temporal tagger capable of processing Spanish and English texts that improves the state of the art in the legal domain, (3) an event extractor for European legal decisions that also generates a timeline, and (4) a pipeline that allows transforming European legal decisions into a set of events within a knowledge graph. For this purpose, several tools and resources have been developed, such as an ontology that allows representing a document as an aggregation of its most relevant events and its temporal annotations, or a converter between different temporal annotation formats and data conforming to this ontology. All these contributions allow to transform a legal document into an event-based representation that facilitates retrieving legal information.

Resumen

Los documentos legales pueden llegar a ser largos, complejos y difíciles de entender. No obstante, existe una fuerte demanda de acceso a información legal, y diariamente se publica una gran cantidad de documentos. Pese a que existen multitud de portales institucionales a disposición de ciudadanos y profesionales del derecho, los documentos en sí suelen ser texto plano de los cuales es difícil extraer información. La recuperación de información temporal en las sentencias judiciales es especialmente importante, y el análisis de estos textos requiere a menudo identificar fechas y eventos. De hecho, poder representar una sentencia como un conjunto de eventos relevantes sería extremadamente útil, pues permitiría mejorar las búsquedas y facilitar la visualización y comprensión de los textos mediante resúmenes y líneas temporales. Sin embargo, no existe a día de hoy un sistema que facilite el procesamiento de información temporal en documentos del ámbito legal.

Esta tesis doctoral contribuye al avance del estado del arte proporcionando un marco de trabajo que aborde la información temporal de manera integral, proponiendo algoritmos de reconocimiento de expresiones temporales y eventos, describiendo un sistema de representación de los mismos y demostrando que su uso facilita consultar información temporal en textos jurídicos.

Las principales contribuciones de esta tesis son (1) diversos corpus anotados en el dominio legal, (2) un anotador temporal capaz de procesar textos en español e inglés que mejora el estado del arte en el dominio legal (3) un extractor de eventos para sentencias europeas que genera además un timeline, y (4) un pipeline que permite transformar sentencias europeas en un conjunto de eventos dentro de un grafo del conocimiento. Para ello se han desarrollado distintos recursos, como una ontología que permite representar un documento como sus eventos más relevantes y sus anotaciones temporales, o un conversor entre distintos formatos de anotación temporal y los datos representados conforme a la ontología. Estas aportaciones permiten una representación del documento que facilita el acceso a la información.

Contents

List of Figures

List of Tables

Part I

INTRODUCTION

Chapter 1

Introduction

"*What then is time? If no one asks me, I know; if I want to explain it to a questioner, I do not know.*"
— Saint Augustine, Confessions

Time has always been fundamental in human life. It is how we measure existence, the dimension we use every day to ubicate events and plan the next steps. Scientists and philosophers, among others, have studied it from ancient times, modelizing it and making it understandable and affordable to humankind. The concept of time is also intrinsic to society: different cultural backgrounds and needs lead to different perceptions of time. As analyzed by Lena Boroditsky (Boroditsky, 2011), language has a deep impact on how humans represent time, defining it as follows:

> "*Time is a topic of central interest in our culture. (...) Time is ubiquitous and yet ephemeral. It forms the very fabric of our experience, and yet is unperceivable: we cannot see, touch or smell time.*"

Regarding representation, Boroditsky concludes that space is the most intuitive realization for humans.

> "*To represent time, people around the world rely on space. We specialize time in cultural artifacts like graphs, time-lines, orthography, clocks, sundials, hourglasses, and calendars; we gesture temporal relations, and rely heavily on spatial words (e.g., forward, back, long, short) to talk about the order and duration of events.*"

The work of Boroditsk analyzes how different languages deal with the concept of time. Some cultures are more likely to represent time as a vertical axe than as a horizontal one; this is the case for Chinese and English speakers, respectively. This time-line steadiness also changes depending on the language; while Chinese people tend to express time-moving metaphors, in English expressions is the person who moves along the time-line. To refer to parts of this time-lines, some languages (such as modern Greek) use amount expressions (πολύωραη) , literally "much time"), while others refer to lengths (English, "a long time"). Similarly, the *sense* of time varies depending on the language: English people, for instance, consider that future is *ahead* of us, while Aymaras consider it is *behind* us.

In short, time is something ethereal, difficult to understand, and whose interpretation depends to a great extent on culture. This is why throughout history mankind has developed different systems to measure and organize time, such as calendars. Calendars are a standard way of tracking time, and being able to assimilate time-related expressions into a fixed system such as these is fundamental to natural language processing.

1.1 Time in Natural Language Processing

In the frame of Natural Language Processing, a *temporal tagger* is a information system that extracts temporal expressions from texts and recognizes their meaning. *Time expressions* (also known as *temporal expressions* or *TEs*) are "constructions referring to points or intervals on the timeline" (Saurí et al., 2010), and in general they can be understood as *anything that answers the questions 'when' or 'how long' but does not involve an event* (e.g. "2 May 2019" or "one hour"). Temporal taggers must first identify the time expressions (*identification*), and then resolve (*normalization*) their meaning, obtaining a fixed date from expressions such as "tomorrow". Table 1.1 shows some examples of normalization.

Additionally, temporal information encompasses not only temporal expressions but also events. Almost anything can be considered an event in a text, and delimiting what to extract and what is the relevant contextual information (e.g. who, how, why) is not trivial. Also categorization of events is tricky and extremely subjective, hindering the automation of its extraction and exploitation for later tasks.

#	Spanish Expression	English	Normalized Value
1	mañana	tomorrow	2019-12-21
2	el mes que viene	the next month	2020-01
3	el pasado lunes	last Monday	2019-12-16

Table 1.1: Examples of normalization for several time expressions using as reference date (*anchor date*) December 20, 2019 (Friday).

However, in many cases in the Natural Language Processing literature dates have been assimilated as named entities, and other temporal expressions have been directly ignored. In the case of events, they are usually extracted *ad hoc* to specific tasks, so there are as many definitions of events as purposes exist. Temporal Tagging has been relegated in NLP as a very specific task with hardly any practical use, being exploited only for academic knowledge in certain types of texts for challenges (mainly news). Only in certain areas such as the medical domain is there extensive literature on clinical temporal tagging and event extraction including standards, resources, and practical application.

Yet there is another domain in which Natural Language Processing (NLP) has not been extensively exploited and which by its nature would benefit greatly from temporal information processing: the legal domain.

1.2 Temporal Information in the Legal Domain

There is a strong demand for access to legal information. The European Union portal EUR-Lex, which publishes legislation, case law, and other documents related to EU law, serves more than two million documents monthly[1]. Nonetheless, the opinion barometers of the CGPJ (Consejo General del Poder Judicial, the Spanish General Council of the Judiciary) show that 82% of citizens consider that legal language is excessively complicated and difficult to understand (Comisión de expertos Modernización del lenguaje jurídico, 2011). This language, commonly named *legalese*, notably hinders the processing task.

[1] `https://eur-lex.europa.eu/statistics/usage.html`. Unless explicitly mentioned, all the URLs were last checked on October 2021. If they were not available back then, the last available version from the Internet Archive (`https://web.archive.org/`) is provided.

The information demanded by the user of this and other search portals often focuses on named entities, such as persons, companies, or dates, which have to be extracted from the raw text. The retrieval of temporal information in judgments is especially important, and the analysis of these texts often requires identifying the referenced events. Although this information extraction task is now usually performed manually, it is possible to devise advanced services such as creating timelines, writing automatic summaries, or advanced querying of documents using better temporal filters.

The provision of these services requires the use of NLP techniques and representation options. However, although there is a discipline specialized in legal text information retrieval (Legal Information Retrieval) with significant advances in each of these areas, it remains an open research topic, and there is not a complete system that facilitates the search for temporal information in text documents in the legal field. This doctoral thesis aims to contribute to the advancement of the state of the art by providing contributions such as a temporal tagger, different annotated corpora that cover an important gap in the domain, and a full pipeline of tools and resources that allow to extract, represent and visualize relevant events in legal texts.

In summary, *when* is as important as *what*. Our age is one of the first things we are taught to say in another language in order to identify ourselves. Time is one of the most basic things we learn, since it is a must for everyday life: when it happens is, in fact, one of the essential coordinates for any event. And yet, let us face it: we are not quite clear on what time is, or even if it exists. Physicists can't agree on something as basic as a definition of time. All we know is that there is something: we feel its consequences, we name it, we divide it, and we organize our lives around it. In this thesis, we do not intend to solve the problems of time, but to advance in the way of processing it at the textual level, focusing on texts which are difficult to understand such as those in the legal domain. The ways in which time manifests itself in a text, what each of these ways indicates to us, how to extract knowledge from them. This is the objective of this thesis.

1.3 Structure of the document

This chapter presents the context of the main focus of the thesis, time and events, both in the cultural and the technological point of view, as well as different activities accomplished[2]. The remaining of the dissertation has been organized as follows.

Chapter 2 aims to provide a broad state of the art on four main topics, namely: the representation of temporal information, the related resources available, the existing technologies for its processing, and the different forms of evaluation that exist. The aim of this chapter is to clearly establish the situation of the research at the time of undertaking this thesis.

Chapter 3 introduces the most important challenges identified in temporal information processing, both in general or particularly related to the legal domain.

Chapter 4 outlines the goals and contributions of this thesis, along with the hypothesis and the assumptions defined, as well as the methodology followed.

Part II: Time Expressions

Chapter 5 inaugurates the second part of the dissertation, which provides the results related to temporal expressions. This chapter presents the corpora annotated with temporal expressions created during this Ph.D. thesis.

Chapter 6 describes the work related to the temporal tagging research done during this thesis, mainly a temporal tagger for Spanish and English texts able to process legal texts and that improves the previous coverage for Spanish.

Part III: Events

Chapter 7 is devoted to event processing. This chapter describes the first corpus annotated with events in the legal domain, named EventsMatter.

Chapter 8 presents the tools dealing with event extraction in the legal domain result of this thesis: ContractFrames and WhenTheFact. First, ContractFrames is a very specific framework able to extract events related to the lifecycle of a real estate contract. Secondly, WhenTheFact is a tool able to extract the most relevant events in a judgment from the European Court of Justice or the European Court of Human Rights and represent them in the form of a timeline.

[2]The title of the thesis refers to the different steps in temporal information processing: first, the text is processed (e.g., structure extraction, sentence splitting, or lemmatization); then, the temporal information is identified; finally, the information must be properly represented.

Chapter 9 provides some event-related resources created in order to boost event processing. These resources include an ontology to represent temporal information annotation from different levels of abstraction, a converter between different temporal annotation formats, and a first attempt of an Event-Based Knowledge Graph in the legal domain.

Part IV: Conclusions

Finally, Chapter 10 reviews the main contributions of the thesis, relating them to the objectives outlined in Chapter 4, as well as the conclusions and the future work envisaged.

1.4 Publications

The following sections list the works published during the accomplishment of this Ph.D. thesis. Additionally, Fig. 1.4 shows the timeline of the thesis, including the publications and the research stays done within.

1.4.1 Journal contributions

- Spanish corpora for Sentiment Analysis: a survey. (2019) **M. Navas-Loro**, V. Rodríguez-Doncel. *Language Resources and Evaluation*, pp 1–38.

- TempCourt: evaluation of temporal taggers on a new corpus of court decisions. (2019) **M. Navas-Loro**, E. Filtz, V. Rodríguez-Doncel, A. Polleres, S. Kirrane. *The Knowledge Engineering Review*, Vol 34, E24.

- Annotador: a Temporal Tagger for Spanish. (2020) **M. Navas-Loro**, V. Rodríguez-Doncel. *Journal of Intelligent & Fuzzy Systems 39 (2020)*, Vol 2, 1979–1991.

1.4.2 Conference contributions

- Spanish Corpus for Sentiment Analysis Towards Brands. **M. Navas-Loro**, V. Rodríguez-Doncel, I. Santana-Perez, A. Sánchez. In *Speech and Computer: 19th International Conference, SPECOM 2017, Hatfield, UK, September 12-16, 2017,*

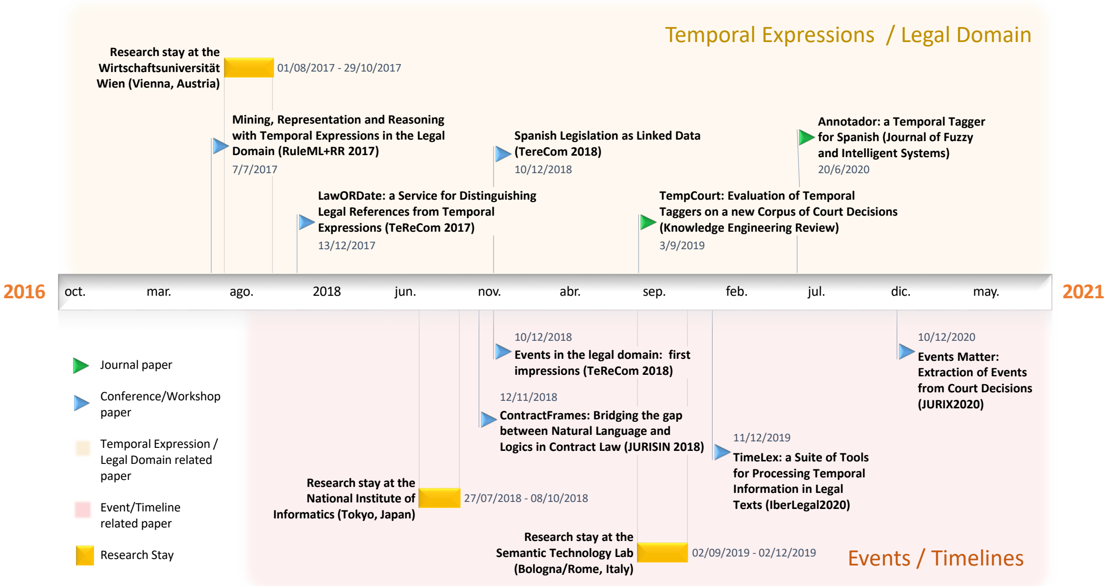


Figure 1.1: Timeline of the Ph.D. thesis, including the research stays done and the publications produced. This timeline does not include submitted papers, just accepted ones. For space reasons, the publications out of the scope of this thesis (mainly the ones related to Sentiment Analysis) are omitted.

Proceedings. Ed. By Alexey Karpove. Cham: Springer International Publishing, pp. 680–689.

- Events Matter: Extraction of Events from Court Decisions. E. Filtz, **M. Navas-Loro**, C. Santos, A. Polleres, S. Kirrane. In *Proceedings of the 33rd International Conference on Legal Knowledge and Information Systems (JURIX 2020), pp. 33-42.*

1.4.3 Workshop contributions

- Mining, Representation and Reasoning with Temporal Expressions in the Legal Domain (2017). **M. Navas-Loro**. In *Proceedings of the Doctoral Consortium, Challenge, Industry Track, Tutorials and Posters @ RuleML+RR 2017 hosted by International Joint Conference on Rules and Reasoning 2017 (RuleML+RR 2017).*

- OEG at TASS 2017: Spanish Sentiment Analysis of tweets at document level (2017). **M. Navas-Loro**, V. Rodríguez-Doncel. In *Proceedings of TASS 2017:*

Workshop on Semantic Analysis at SEPLN (TASS 2017). Ed. by Julio Villena Román et al. Vol. 1896. CEUR Workshop Proceedings. Murcia, Spain: CEUR-WS, pp. 43–49.

- MAS: A Corpus of Tweets for Marketing in Spanish (2018). **M. Navas-Loro**, V. Rodríguez-Doncel, I. Santana-Perez, A. Fernández-Izquierdo, A. Sánchez. In *The Semantic Web: ESWC 2018 Satellite Events. Ed. by Aldo Gangemi et al. Cham: Springer International Publishing, pp. 363–375. ISBN: 978-3-319-98192-5.*

- LawORDate: a Service for Distinguishing Legal References from Temporal Expressions (2017). **M. Navas-Loro**. In *Proceedings of the 1st Workshop on Technologies for Regulatory Compliance co-located with the 30th International Conference on Legal Knowledge and Information Systems (JURIX 2017), Luxembourg, December 13, 2017. pp. 25–31.*

- Events in the legal domain: first impressions (2018) **M. Navas-Loro**, C. Santos. In *Proceedings of the 2nd Workshop on Technologies for Regulatory Compliance co-located with the 31st International Conference on Legal Knowledge and Information Systems (JURIX 2018), Groningen, December 12, 2018.*

- Spanish Legislation as Linked Data (2018) V. Rodríguez-Doncel, **M. Navas-Loro**, E. Montiel-Ponsoda, P. Casanovas. In *Proceedings of the 2nd Workshop on Technologies for Regulatory Compliance co-located with the 31st International Conference on Legal Knowledge and Information Systems (JURIX 2018), Groningen, December 12, 2018.*

- TimeLex: a Suite of Tools for Processing Temporal Information in Legal Texts (2019) **M. Navas-Loro**, V. Rodríguez-Doncel. In *Proceedings of the 2nd Workshop Iberlegal 2019 co-located with the 33rd International Conference on Legal Knowledge and Information Systems (JURIX 2020), Madrid, December, 2019.*

1.4.4 Submitted contributions

The following contributions are currently under review in different journals.

- (SUBMITTED) Lynx: A Knowledge-based AI Service Platform for Content Processing, Enrichment and Analysis for the Legal Domain. (2020) J. Moreno Schneider, G. Rehm, E. Montiel-Ponsoda, V. Rodríguez-Doncel, P. Martín-Chozas, **M. Navas-Loro**, M. Kaltenböck, A. Revenko, S. Karampatakis, C. Sageder, J. Gracia, F. Maganza, I. Kernerman, D. Lonke, A. Lagzdins, J. Bosque Gil, P. Verhoeven, E. Gomez Diaz, P. Boil Ballesteros *Special Issue of the Information Systems Journal.*
- (SUBMITTED) Tools for building an event-based knowledge graph from legal decisions. (2021) **M. Navas-Loro**, V. Rodríguez-Doncel. *Special Issue on Event-centric Open Analytics, Semantic Web journal.*

Additionally, the software *Añotador*, that will be described in Chapter 6, has been presented by Universidad Politécnica de Madrid at the registry of Comunidad de Madrid under registration number M-000922/2021.

1.5 Scholarships

This work has been supported by a grant from the Programa Propio de la Universidad Politécnica de Madrid, from January 2018 to January 2022. From 2016 to 2017, this work was supported by a grant from the Universidad Politécnica de Madrid's Oficina de Transferencia Tecnológica. During part of 2017 this work was supported by a contract as a research assistant from Comunidad de Madrid.

1.6 Research stays

The research stays done in the context of this thesis are outlined below.

01.08.17 – 29.10.17 (3 months) Research stay at the **Wirtschaftuniveritāt Wien (Vienna University of Economics and Business), Vienna, Austria**. In this stay, the TempCourt corpus was created, and the results and the main lacks of existing temporal taggers were analyzed. Also an analysis of the main particularities of the legal domain with regard to temporal annotation was performed. These tasks were done with the supervision of Prof. Dr. Sabrina Kirrane and Prof.

Dr. Axel Polleres, and published as a journal paper (Navas-Loro et al., 2019a). The stay was funded by the Consejo Social de la Universidad Politécnica de Madrid.

27.07.18 – 08.10.18 (2,5 months) Research stay at the **National Institute of Informatics (NII), Tokyo, Japan**, supervised by Prof. Dr. Ken Satoh. During this stage ContractFrames, a first approach to the processing of events in the legal domain (Navas-Loro et al., 2019b), was developed. This stay was funded by the NII International Internship Program.

02.09.19 – 02.12.19 (3 months) Research stay at the **Semantic Technology Lab in Bologna and Rome, Italy**, supervised by Prof. Aldo Gangemi. During this stay, the software FRED (Gangemi et al., 2017) was used to transform natural language in legal judgments into RDF representation including events. Then, the queries that extracted the information related to events, namely its core, the actors involved, and other context information (such as where it happens) were explored. This stay was funded by the Programa Propio de la Universidad Politécnica de Madrid.

1.7 Projects

The following projects framed the work presented in this thesis:

- Legal Knowledge Graph for Multilingual Compliance Services (LYNX), funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 780602.
- LPS-BIGGER, a national project with id IDI-20141259, aimed to perform emotion and sentiment analysis towards brands in tweets.

Chapter 2

State of the Art

This chapter presents the state of the art on temporal information processing and representation. Since there are several tasks involved and the literature is vast, the areas into which this review has been divided will be introduced first, and then the methodology followed will be described. It must also be noted that part of the state of the art presented here was already included in the different publications done during this thesis (Filtz et al., 2020; Navas-Loro, 2017; Navas-Loro et al., 2019a; Navas-Loro and Rodríguez-Doncel, 2020; Navas-Loro and Santos, 2018; Navas-Loro et al., 2019b).

This literature review is conducted to identify the state of the art in the following research areas:

a) Representation options for temporal information

b) Resources for processing temporal information

c) Technologies for processing temporal information

d) How to evaluate the processing of temporal information

For each of these areas, in turn, the existing literature is analyzed for (1) time expressions and (2) events. Since these research topics are too broad, the traditional approach to search papers is unattainable, and integrative literature review methodology was observed (Torraco, 2005; Whittemore and Knafl, 2005). However, in some parts also techniques discussed in Kitchenham and Brereton (2013) were used.

The first step in integrative reviews is the problem definition and the scope of interest (Whittemore and Knafl, 2005). I focused on papers describing (1) reviews on any of the previous research areas, (2) available tools, resources, and representation options, preferably used frequently, (3) theoretical analyses, if they are focused and applicable to the research, (4) challenges describing tools and evaluations.

The following step consists of literature search; although in Whittemore and Knafl (2005) is stated that computerized databases are limited and suggest manual journal checking, I consider this is not applicable to the current days nor to the domain, since computer science conferences and journals tend to publish proceedings and issues online. I, therefore, did the most research on databases, although also "physical" proceedings and books were occasionally consulted. The databases consulted included for instance Web of Science, Science Direct, or Google Scholar. Different keywords related to the tasks tackled in the thesis were used in different stages of the work in order to retrieve as many related works as possible, using also forward and backward snowballing (Kitchenham and Brereton, 2013). Also, networking was extremely useful: colleagues gently sent any pointers they considered of interest, and also reviews of submitted papers included valuable references to my research.

Additionally, some methodologies (Kitchenham and Brereton, 2013) suggest identifying journals or conferences to search papers on the topic. Nevertheless, NLP conferences tend to be very heterogeneous, and the only sources available were old challenges (e.g. TempEval[3]) and the TIME symposium[4] (mostly oriented to temporal reasoning), as well as occasional workshops (e.g. EVENTS[5]). Finally, in the data analysis stage, the different papers found were cribbed, organized, and clustered in groups that constitute the different subsections in the present state of the art.

Regarding the state of the art organization, it will be as follows. First, the different options available for temporal information representation will be presented (Section 2.1). Section 2.1.1 will introduce TimeML, the ISO standard usually employed in temporal processing, analyzing the concepts and the tasks involved in TimeML. Then other alternatives will be presented, including annotation schemas and ontologies that cover time-related concepts (Section 2.1.2) and events (Section 2.1.3).

To follow, the resources available when processing temporal information will be presented (Section 2.2), including corpora (Section 2.2.1) and semantic resources (Sec-

[3]https://web.archive.org/web/20200811064918/https://www.cs.york.ac.uk/semeval-2013/task1/

[4]http://time.di.unimi.it/TIME_Home.html

[5]https://www.aclweb.org/anthology/venues/events/

tion 2.2.2), both generic resources related to temporal information (Section 2.2.2.1 and Section 2.2.2.2) and those specific to the legal domain (Section 2.2.2.3).

Afterwards, Section 2.3 presents the technologies related to temporal information processing. First, a number of generic temporal taggers available in the literature that have been tested and evaluated (Section 2.3.1) are briefly described. Next, some proposals in the literature about temporal expression processing in the legal domain (Section 2.3.2) are introduced. Subsequently, some Named Entity Recognition (NER) tools capable of partially detecting temporal expressions, which are considered as another Named Entity (Section 2.3.3), will be reviewed. Lastly, proposals for event processing (Section 2.3.4), both generic (Section 2.3.4.1) and specific to the legal domain (Section 2.3.4.2) will be revised, including technologies not specifically oriented to temporal information processing but performing partially similar tasks (Section 2.3.4.1).

Finally, how temporal information extraction has been evaluated in different scenarios will be briefly revised in Section 2.4.

2.1 Representation of Temporal Information

Several schemas and standards have been proposed in literature for the annotation and representation of temporal information. Whereas some of them are generic, describing temporal information without targeting a specific domain, some of them usually try to cover the needs of different tasks, focusing on different aspects and emphasizing the features that are required for a specific use case. Among all of them stands out the TimeML ISO standard, the most widespread ISO time-focused mark-up language used for temporal annotation.

The organization of this section will be as follows. Section 2.1.1 briefly introduces this TimeML annotation standard and encompasses the concepts it handles and the tasks for which it is used. Subsequently, other representation alternatives for temporal expressions will be presented, mainly annotation schemas (Section 2.1.2). Finally, event representation alternatives to TimeML, including annotation schemas and ontologies, will be described in Section 2.1.3.

2.1.1 TimeML

TimeML is an annotation scheme specifically designed for the markup of events, times, and their temporal relations in text (Pustejovsky et al., 2010), recently converted into an ISO standard[6]. Times are marked using the TIMEX3 tag, which is based on the TIMEX tag of the Sheffield Temporal Annotation Guidelines (STAG) (Setzer, 2002) and in the TIMEX2 tag of the TIDES ACE TIMEX2 standard (Ferro et al., 2005).

2.1.1.1 Concepts considered by TimeML

The TimeML standard covers different types of temporal information.

- **Temporal Expressions** are *"constructions referring to points or intervals on the timeline"* (Saurí et al., 2010). The tag `TIMEX3` is used for marking them in the text, and they can be classified in four types:
  - DATE: calendar dates from different granularities: years, dates, seasons, quarters, etc. It also includes references to the past, present, and future. Examples of DATEs are *tomorrow, January 22, last month, the 60's, this fall,* or *now.* It is also used to express key dates of a document, such as its creation or publication date.
  - TIME: used for day times that are smaller than a day, such as *midnight, half past six, eight in the morning, 5 p.m.* or *6.45 Monday, August 19, 2019.*
  - DURATION: denoting the lasting of something. It can be quantified over a time unit (*six years, half an hour, a decade, a fortnight*) or fuzzy (*a lot of time, a while*).
  - SET: applied to repetitive time expressions, such as *annually, three times a month, every Tuesday* or *two days a week.*
- **Relations** are connections between events, temporal expressions, or a mixture of both. The TimeML standard includes three types of relations:

[6]ISO 24617-1 Language Resource Management - Semantic Annotation Framework (SemAF) - Time and Events (SemAF Time and ISO-TimeML)

- TLINK: temporal relation between events, times, or between an event and a time. This relation can be[7]: *simultaneous, before (after), immediately before (immediately after), including (being included), during (being held during), beginning (begun by), ending (ended by), identity, set/subset.*
- SLINK (Subordination Link): it relates two events. It can be modal, factive, counter-factive, evidential, negative evidential, or conditional.
- ALINK (Aspectual Link): it relates aspectual verbs like '*start* to do something' to their argument.

- **Events** are, regarding the TimeML guidelines (Saurí et al., 2009; Saurí et al., 2006a):

  > *(...) situations that happen or occur. Events can be punctual or last for a period of time. We also consider as events those predicates describing states or circumstances in which something obtains or holds true (...)*

  Events can be expressed as verbs, nominalizations, adjectives, predicative clauses, or prepositional phrases. They are marked using the tag *EVENT* and can be classified into one of the following categories:

  - REPORTING events are those where a part communicates something, and includes verbs such as *say, explain, state* or *cite.*
  - PERCEPTION events are those that imply physical perception of another event, including verbs like *listen, hear* or *view.*
  - ASPECTUAL events are those grammar verbs used to denote the different phases of an event, namely *initiation* (*begin, set out, originate*), *reinitiation* (*restart, reinitiate*), *termination* (*stop, end, abandon, block*), *culmination* (*finish, complete*) and *continuation* (*go on, proceed, persist, persevere*).
  - I_ACTION events (stands from *Intensional Action*) introduce an event argument while providing some extra information about that event. This class encompasses events like *attempt, investigate, delay, avoid, persuade, offer, swear, appoint* or *claim.*

[7] The relation in brackets is the inverse relation.

– I_STATE events (stands from *Intensional State*) are states that refer to alternative worlds. It could be said that they are similar to I_ACTIONs but for states. Examples of this type are *believe, be sure, desire, hope, hate, require, be prepared*, and *be able.*

– STATE events describe circumstances for something to be true.

– OCCURRENCE events are those describing events that happen or occur but are not covered by the previous classes.

Additionally, the TimeML guidelines include SIGNAL and MAKEINSTANCE tags.

SIGNALs are text elements that make explicit relations between pieces of temporal information, such as temporal prepositions (such as *at, before* or *during*), temporal conjunctions (*while* or *when*), prepositions signaling modality (*to*) or special characters we can find in temporal expressions (e.g., '-' or '/').

MAKEINSTANCE tags are the realisations of events. While the EVENT tag will mark the textual mention of an event, MAKEINSTANCE is a tag out of the text that represents each actual happening of the event, the instances of it. The following example of the guidelines (Saurí et al., 2006a) facilitates discerning this distinction:

John taught on Monday and Tuesday.

While in the text *taught* would be marked as an EVENT, it can be derived that there are actually two instances of it, one on Monday and another one on Tuesday. Two MAKEINSTANCEs would therefore be annotated.

2.1.1.2 Tasks considered by TimeML

The functionalities of temporal taggers based on the TimeML standard can be classified into four categories; as shown in Table 2.1, some temporal taggers support all functionalities, while other taggers require some additional tools.

- *Identification* task consists of detecting temporal expressions in a text, marking their extent correctly with regard to the TimeML standard.

- *Normalization* task involves giving a standard value to each temporal expression. In TimeML, this value is determined by the ISO 8601 norm. Sometimes this *normalized* value can be directly derived from the temporal expression (e.g., in the sentence "*On 25 April 1945, Italian partisans liberated Milan and Turin.*", where the underlined temporal expression normalized value would be '1945-04-25'), and sometimes it has to be *anchored* to a specific date. Some examples of normalization can be found in Table 1.1, extracted from Navas-Loro and Rodríguez-Doncel (2020).

- *Event extraction* task identifies the events in a text, classifies them among several types of events, and gives them a value for each attribute.

- *Relation extraction* task tackles the identification of *TLINKS, ALINKS* and *SLINKS* in a text. It is the least implemented task.

2.1.2 Time Expression representation alternatives to TimeML

Although TimeML is the most widely used standard for temporal information annotation, there are other options for representing time, both as annotations and in the form of ontologies.

Among annotation standards, we find for instance TIDES TIMEX2 (Ferro et al., 2001), in which were based TIMEX3 tags from TimeML. Although there exist some corpora annotated with TIMEX2 tags, nowadays this format is no longer used. Other general-purpose annotation standards can also be used to represent TEs, such as the W3C Web Annotations[8], the NLP Interchange Format[9] (NIF) (Hellmann, 2012) or NLP Annotation Format (NAF)[10]. These formats are not specifically designed for temporal information representation, but they support information for NLP annotations. We can also find in literature extensions of TimeML for specific domains, such as the medical extension done for the THYME project (Styler et al., 2014), or alternatives such as the probabilistic approach to values proposed by Angeli et al. (2012), that instead of normalizing with a fixed value for a temporal expression suggested using a range of probabilities (that is later assimilated to a single value for testing).

[8] `https://www.w3.org/TR/annotation-model/`

[9] `http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#`

[10] `https://github.com/newsreader/NAF`

Regarding ontologies, some of them will be presented in the next section, since usually ontologies representing events also represent time. In any case, a very complete overview of time-related ontologies, including an analysis of the main features to consider when choosing one for a specific task, is provided by Ermolayev et al. (2014). Similarly, an example of a review of time-related ontologies for a specific domain application, namely semantic web, is that of Fernández-López and Gómez-Pérez (2004).

2.1.3 Event representation alternatives to TimeML

While temporal expressions have been often subsumed to Named Entities, or treated from a philosophically oriented perspective more than from a practical one, event representation has received plenty of attention in literature, especially in the form of event-related ontologies.

Ontologies, on the one hand, cover time-related information from a top approach. This is, they facilitate classes to represent different aspects relevant to temporal information, but do not tend to go deeper on each of their realizations in the real world, but to handle just abstract information about them. Annotation schemas, on the other hand, tend to focus on detecting appearances of certain predefined temporal information, such as event taxonomies and their arguments in texts. They, therefore, specify subtypes and expected arguments for each kind of event, admitting also other information per event instance, such as its probability or factuality. To summarize this idea, we could say that ontologies offer a more flexible and abstract representation option, while annotation schemas have a more strict and predefined target, oriented to a real-world use case.

2.1.3.1 Annotation schemas

The main source of event-related annotation schemas is the different challenges carried out in the last years. Despite this means that in most cases also a corpus was annotated, these corpora are usually not available for free, hindering their reuse out of big companies. The main event annotation schemas are summarized below. Very useful brief analyses comparing most of them have been performed in literature (Aguilar et al., 2014; Ahn, 2006).

ACE

ACE (ACE English Annotation Guidelines for Events, 2005) is an annotation schema that has its own definition of event[11]. The ACE project guidelines focus on different types of events, namely LIFE, MOVEMENT, TRANSACTION, BUSINESS, CONFLICT, CONTACT, PERSONNEL, and JUSTICE. In the subtypes of the latter we find *arrest-jail, release-parole, trial-hearing, charge-indict, sue, convict, sentence, fine, execute, extradite, acquit, pardon* and *appeal*.

ERE

ERE (Entities, Relations, and Events)[12] was an annotation task in the DEFT program that, as its name suggests, include the annotation of entities, relations, and events, as well as their attributes, according to a taxonomy. There are two versions of ERE, named *Light* ERE and *Rich* ERE.

Light ERE is basically a lighter version of ACE aimed to make annotation easier and more consistent (Bies et al., 2016). This simplification includes for instance tagging just actually happening events or not including subtypes of entities.

On the opposite, *Rich* ERE (Song et al., 2015) expands *Light* ERE incorporating more types and subtypes of events (and re-classifying part of those in *Light* ERE), and annotates also future, hypothetical or conditional events. Additionally, the concept of *event hopper* is included as a more inclusive, less strict notion of event coreference.

KBP Event nugget annotation

The Knowledge Base Population (KBP) was a track framed in the Text Analysis Conference (TAC) that targeted the extraction of different NLP entities, such as events, including coreference. The edition of 2017 also included a Event Sequence task, aimed

[11] *"An event is a specific occurrence involving participants. An Event is something that happens. An Event can frequently be described as a change of state."*

[12] The ERE original guidelines are no longer available online, so the information presented is gathered from publications referring it.

to retrieve the chronological order of events, while later KBP editions shifted to a Streaming Multimedia target, being therefore renamed to SM-KBP[13].

In its 2014 edition, KBP introduced the concept of *event nuggets* (defined as a semantically meaningful unit that expresses the event in a sentence) to annotate events in text (Mitamura et al., 2015), while in the following year edition this definition was redefined to "*the smallest, contiguous extent of text (usually a word or phrase) that most saliently expresses the occurrence of an event*" (Song et al., 2016). The intention of this new concept was two-folded. First, allowing the tagging of multi-word events, since they considered that an event nugget can be either a single word or a continuous or discontinuous multi-word phrase, differently to some previous approaches. Second, to allow discontinuous tagging, which had been a nightmare for annotators in previous related tasks. Each *event nugget* has a trigger (its text span), the event type, the related arguments, and the REALIS value (that indicates if an event actually occurred).

This is the most recent annotation proposal, and it is based on previous efforts - namely Light ERE, since it shares the same 33 event types and subtypes with it. One of the main contributions of *event nuggets* is the analysis of different real use cases of event annotation (Mitamura et al., 2015), such as if non-explicit events should be derived (e.g. in "*Two other* ***assailants*** *have committed suicide.*", the term in bold implies that there was an assault), or if some expressions are actual events or results of events (e.g., the mention of "injuries" in a text suggests there was a previous event that produced them).

Other proposals

Besides the exposed above, there are also other proposals in literature, such as Richer Event Description (RED) (O'Gorman et al., 2016), old challenges like MUC, that included a task named Scenario Template Task where information about an event should be extracted (Marsh and Perzanowski, 1998), or alternatives used to annotate corpora like OntoNotes[14] and FrameNet (that will be presented in Section 2.2.2.1).

[13] `https://tac.nist.gov/2019/index.html`

[14] `https://catalog.ldc.upenn.edu/LDC2013T19`

Domain-specific events annotation

Besides the previous efforts usually focused on news annotation and covering a wide scope of events, we also find efforts targeting specific domains. In this line, the work by Sprugnoli and Tonelli (2019) presents annotation guidelines to mark up flexible extents of historical events and classify them following a proprietary taxonomy adapted from the Historical Thesaurus of the Oxford English Dictionary (Kay et al., 2009).

There are also initiatives, such as the Open Event Date Alliance[15], that focus on parsing events from news sources and generating replicable data from them, instead of annotating them in text. Inside the wide universe of news-oriented event extraction or annotation, we also find more specific use cases, such as protest-event representation options like the CAMEO ontology (Schrodt, 2012), that are often based on previous approaches. Usually, these efforts require full projects or Ph.D. thesis to be properly analyzed and tackled (Danilova, 2015).

Additionally, sometimes several ontologies are used at the same time in order to annotate events. This is the case for instance of the GAF annotation framework for Events[16], that relies on the SEM ontology and TAF (TERENCE Annotation Format), the latter being at the same time based in ISO-TimeML and adapted to cover children's stories events. GAF was used in the NewsReader European project, aimed to build structured event indexes of large volumes of financial and economic data for decision making[17].

To summarize, when dealing with event annotation there are two main approaches. The TimeML approach, on the one hand, is very linguistic-oriented and just links events in the text to other temporal information (both temporal expressions or other events). This allows to cover all event mentions, but constraints the information that can be related to them. On the other hand, challenges such as ACE provide a series of templates for annotation, predefining the information to be found in the text, such as arguments or roles. This allows to store more information, but of course, leaves a lot of not considered events aside.

[15] `https://openeventdata.github.io/`
[16] `http://groundedannotationframework.org/`
[17] `http://www.newsreader-project.eu/the-project/`

2.1.3.2 Ontologies and data models

The main available ontologies modeling events and time are briefly reviewed below since they usually come together. There are more ontologies dealing with events, such as Model F (Scherp et al., 2012), but due to the extensive literature on the topic this account analyses only the most related and well-known proposals.

PROV In the PROV Data Model, an activity is "*something that occurs over a period of time and acts upon or with entities; it may include consuming, processing, transforming, modifying, relocating, using, or generating entities*"[18]. Although PROV is not specifically designed for representing events and time-affairs, it is commonly used together with the Time Ontology for this purpose.

W3C Time Ontology Recommendation The Time Ontology[19] is the most well-known ontology for representing time and provides the means for anchoring events in time. It represents dates, durations, intervals, and temporal relations. As mentioned before, it is often used together with the PROV ontology. In fact, the class *TemporalEntity* can be considered as a superclass of prov:Activity, while *Instant* would be prov:InstantaneousEvent's.

Simple Event Model SEM[20] is an ontology created to model events in various subject domains, such as history, cultural heritage, geography or multimedia. The four core classes in this ontology are *Event* (to record what happens), *Actor* (who or what participated in the event), *Place* (where did it happen), and *Time* (when did it happen). Fig. 2.1 shows the main classes of the SEM ontology. Latter efforts in event representation have been done over SEM, and include results such as the EventKG schema (depicted in Fig. 2.2), used in EventKG, a multilingual event-centric temporal knowledge graph that incorporates over half a million contemporary and historical events (Gottschalk and Demidova, 2018a).

[18] `https://www.w3.org/TR/prov-dm/#term-Activity`
[19] https://www.w3.org/TR/owl-time/
[20] `https://semanticweb.cs.vu.nl/2009/11/sem/`

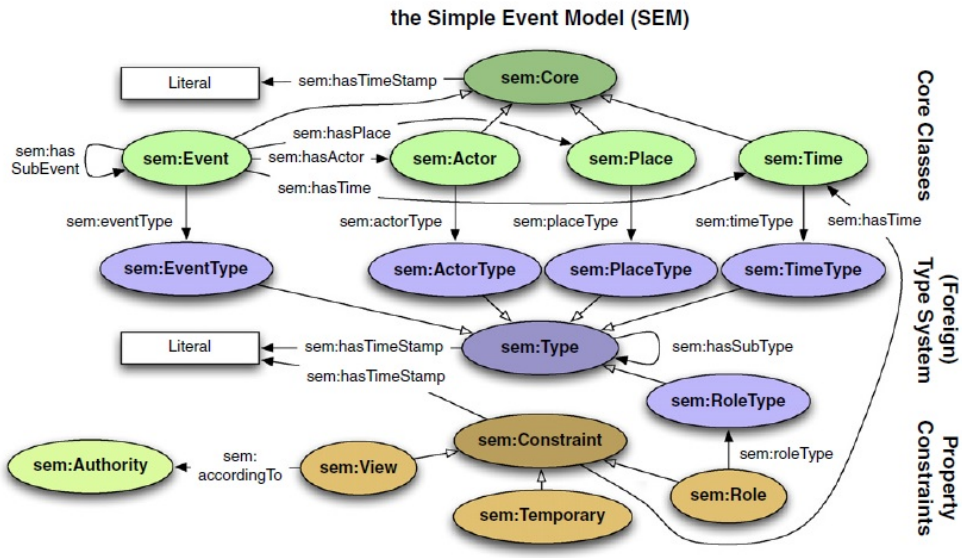


Figure 2.1: Event classes from the SEM Ontology. Image taken from its documentation.

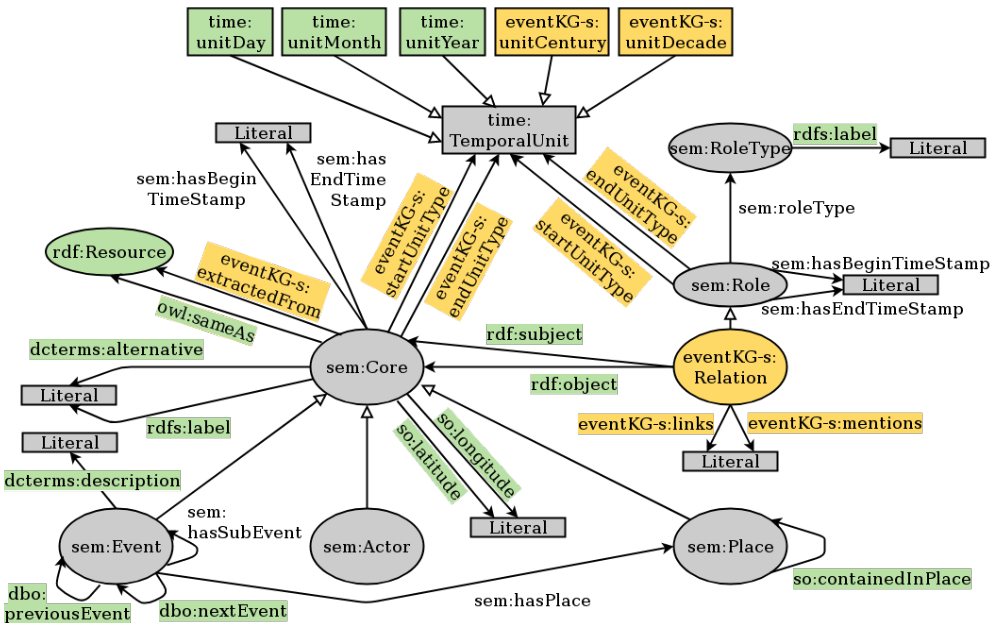


Figure 2.2: EventKG schema, used for building an event-centric knowledge graph. Image taken from its website.

Time Event Ontology TEO[21] is an ontology that allows to represent different temporal information items for the purpose of further reasoning (Li et al., 2020). As it is for the medical domain, subclasses of event such as Clinical Intervention or Patient Accident are covered. Nevertheless, the time-related part of the ontology is very rich in terms of terms for temporal expressions. Since the ontology has many classes, only part of them is shown in Fig. 2.3 to illustrate this richness.

Event Ontology and Timeline Ontology The Event Ontology of Yves Raimond, developed by the Centre for Digital Music in Queen Mary, University of London[22] (2004), provides a basic and flexible representation for a general event despite being conceived in the frame of musical events. It relies on the formal definition provided by (Allen and Ferguson, 1994), namely the following:

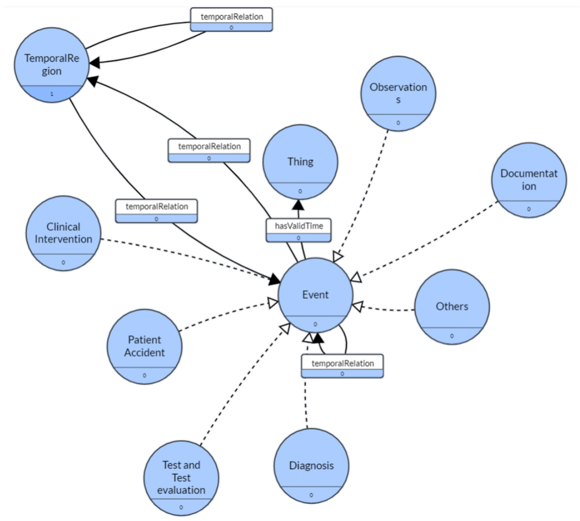
> *"We take the position that events are primarily linguistic or cognitive in nature. That is, the world does not really contain events. Rather, events are the way by which agents classify certain useful and relevant patterns of change."*

Figure 2.4a illustrates the basic classes in this ontology, that has been later complemented with the creation of the Timeline Ontology[23], depicted in Figure 2.4b.

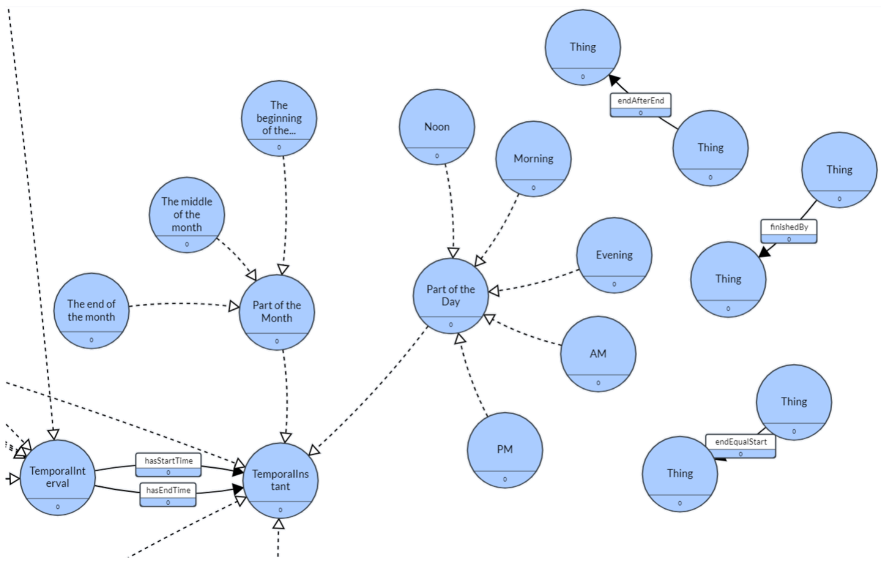
Event and Implied Situation Ontology ESO (Segers et al., 2016) is a manually constructed resource that formalizes the events and the implied situations before, during, and after it, as well as the roles of the entities affected by it[24]. This resource is fully mapped to the SUMO ontology[25] at class level and to FrameNet at class and role level. It was developed together with the Circumstantial Event Ontology for Calamities (CEO)[26]

As exposed above for the most relevant ontologies dealing with events, each option has its own definition of event. Nevertheless, they all tend to include the same classes,

[21] `https://sbmi.uth.edu/bsdi/TEO_1.0.0.owl`
[22] `http://motools.sourceforge.net/event/event.html#`
[23] `http://motools.sourceforge.net/timeline/timeline.html`
[24] `https://github.com/RoxaneSegers/ESO-Ontology`
[25] `http://www.ontologyportal.org/`
[26] `https://github.com/RoxaneSegers/CEO-Ontology`

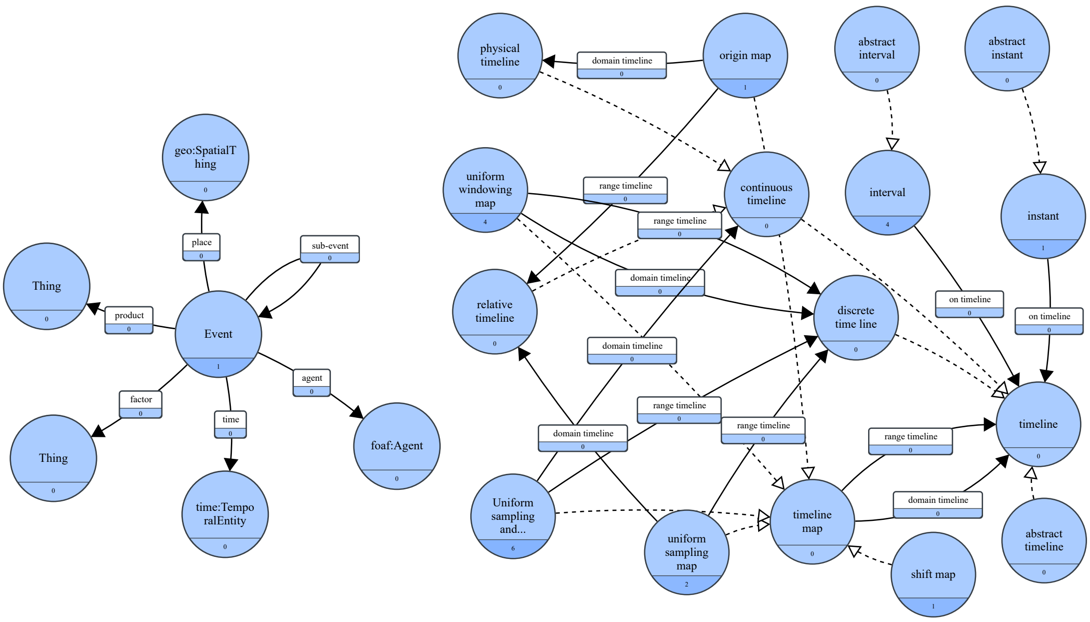


(a) Event class in TEO.



(b) Some classes related to time.

Figure 2.3: Event class and time-related classes from TEO. Images created by the author using `Gra.fo` software.



(a) Event Ontology main classes.

(b) Timeline Ontology main classes.

Figure 2.4: Main classes of the Event and Timeline Ontologies by Yves Raymond. Figures done using the `Gra.fo` software.

introducing just slight differences and additional classes, usually to cover specific use cases. Additionally, besides full-fledged ontologies we also find in literature the Ontology Design Patterns[27] (Krisnadhi and Hitzler, 2017), that suggest how to represent events for different kinds of situations, and the Grounded Annotation Framework GAF, an annotation framework that provides an RDF representation to link instances to *instance mentions* (mainly designed for representing *event mentions*).

2.1.3.3 Event Representation in the Legal Domain

Regarding event representation in the legal domain, one of the most well-known upper ontologies in the legal domain is LKIF, which stands for Legal Knowledge Interchange Format (Gordon, 2008), including more than 200 classes. In LKIF, events are considered *changes* that "*occur against this canvas of temporal and spatial positions*" (Hoekstra et al., 2009). At a phrase level, "events" are represented, and it further provides for the antecedents and consequences of events. Other important concepts herein are actors, objects, time, locations, trades, and transactions, among others. Also, statements are classified into facts and norms.

LegalRuleML [28] is a format for expressing and inferencing over legal knowledge for which Gandon et al. (2017) proposed an extension that supports modeling of normative rules. It does not model events *per se*, but only temporal dimensions of the norms. The concept of event is introduced at the level of phrases. Other concepts are participants, time, locations, jurisdictions, artifacts, and compliance. Participants may be designed as agents, bearers, or third parties, who may have roles and be part of an authority.

Finally, the Oasis standard Akoma Ntoso [29] has become widely known in the last years. Akoma Ntoso is an XML markup schema for describing legal resources of various types, for example, laws, regulations, and court decisions. Events are considered "*actions and occurrences*", although they are not specifically targeted and are considered "*other concepts*"[30].

[27] http://ontologydesignpatterns.org/wiki/Community:Event_Processing
[28] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legalruleml
[29] http://www.akomantoso.org
[30] http://www.akomantoso.org/?page_id=47

Discussion

This section presented the options available for the annotation or representation of time-related or legal information. While for temporal expressions the clearest choice is the TimeML annotation standard, events have so many alternatives available that there is no definite settled option. Additionally, the possibilities when it comes to representing legal information do not usually cover temporal considerations, so there is a need to ease the transition from the temporal annotation standard to a representation that is more easily embraceable in the legal domain.

2.2 Time-related and Legal-related Resources

This section presents the main resources related to temporal expressions and events. Section 2.2.1 introduces some of the available collections of documents including temporal information out of the legal domain since there are no time-related annotated corpora available in this field. Section 2.2.2, on the other hand, reviews other semantic resources related to the tasks covered in this thesis and frequently used in previous approaches in the literature.

2.2.1 General Corpora

Regarding corpora annotated with temporal information, different datasets have been released in challenges, such as the previously mentioned TempEval, or proposed in literature. A more thorough exploration of these corpora reveals that not only the ISO standard TimeML (Pustejovsky et al., 2003a) is used to annotate the expressions, but also other formats, such as TIDES TIMEX2 (Ferro et al., 2001), or simply variations of TimeML, such as the medical extension done for the THYME project (Styler et al., 2014).

An analysis of available corpora shows that some domains and types of text received more attention than others. Most corpora are built from news (e.g., the Timebank corpus (Pustejovsky et al., 2003b), the TempEval challenges datasets, and the MEANTIME corpus (Minard et al., 2016)). Historical texts and medical texts, like the Wikiwars corpus (Mazur and Dale, 2010) and the THYME corpus (Styler et al., 2014) respectively, have also been annotated in the past. Regarding language registers, corpora with scientific abstracts (Strötgen and Gertz, 2012), tweets (Tabassum et al.,

2016) and colloquial texts (Strötgen and Gertz, 2012) can be found. Nevertheless, all the previously mentioned corpora (except the MEANTIME and TempEval datasets, which are multilingual) are exclusively composed of English texts. A very good reference to see the evolving interest in events and how their representation shaped over time is the timeline by Sprugnoli and Tonelli (2017).

Spanish corpora are scarce, and to the best of the author's knowledge, there are only three datasets available for this language with TIMEX3 tags. The Spanish TimeBank corpus[31] (with news and fiction texts), the ModeS TimeBank 1.0[32] (texts from the 17th and 18th centuries) and the MEANTIME corpus (news). There were also Spanish challenges in TempEval 2 and TempEval 3 competitions, but they were built on texts from a task-adapted fragment of TimeBank; additionally, the latter's test dataset is not available online anymore. The Spanish available corpora are therefore scarce and not heterogeneous, notably hindering the temporal tagging task in this language.

Furthermore, works in other fields than the legal one, such as medical, are interesting since both domains share common requirements, such as the need for domain knowledge for identifying specific events and for dealing with the existence of several timelines in the same text, among others.

2.2.2 Related resources

Besides corpora, other resources can be helpful when processing temporal information in texts. Below a brief review of some of these resources is presented, organized according to whether they are related to time, to events, or to the legal domain.

2.2.2.1 Time related

TempoWordNet TWn (Dias et al., 2014) is a free lexical knowledge base for temporal analysis where each synset of WordNet is automatically time-tagged with four dimensions: atemporal, past, present, and future. The aim is to provide a "temporal value" to sentences through the value of the words in them. There are three different resources[33], one trained with semantic considerations (TWnL-1.0), another one with a probabilistic approach (TWnP-1.0), and finally one following a hybrid approach (TWnH-1.0).

[31] `https://catalog.ldc.upenn.edu/LDC2012T12`
[32] `https://catalog.ldc.upenn.edu/LDC2012T01`
[33] `https://tempowordnet.greyc.fr/download_TWn.html`

Time Intervals Time Intervals is a linked data resource from the UK data portal[34] that consists of every time interval and instant into the past and future, from years down to seconds, being, therefore, an infinite set.

2.2.2.2 Event related

The Comprehensive Event Ontology CEVO (Shekarpour et al., 2019) is a conceptualization that provides more than 230 classes for over 3,000 English verbs. It is based on a concept of class that clusters sets of semantically coherent verbs with similar syntactic behaviour.

FrameNet FrameNet (Baker et al., 1998) is a repository of the implementation by Charles Filmore of the concept of semantic frames, described as "*a script-like conceptual structure that describes a particular type of situation, object, or* ***event*** *along with its participants and props*" (Ruppenhofer et al., 2006). Following this definition, 1224 frames[35] on different topics with related linguistic (such as POS tagging, lexical units, and annotated examples) and circumstantial information (such as the arguments related to the situation) are available for academic research.

VerbNet VerbNet (Kipper et al., 2006) is the largest network of verbs in English. It is domain independent and includes both syntactic and semantic information for different verb classes, as well as the thematic roles[36] related (similar to the arguments in FrameNet).

PropBank PropBank (Palmer et al., 2005) was initially a corpus annotated with semantic roles that eventually also derived into a repository of frame files. Each of these files is named after a verb and contains information about the related roles of each of its senses and annotated examples. The repository is open[37] and linked to other initiatives.

[34] https://old.datahub.io/dataset/data-gov-uk-time-intervals
[35] https://framenet.icsi.berkeley.edu/fndrupal/current_status, visited on May 5, 2021.
[36] https://verbs.colorado.edu/verbnet/
[37] http://propbank.github.io/

Some of the previous initiatives have been integrated into a proposal called the Unified Verb Index (UVI)[38], where a single online search gathers results from different repositories.

2.2.2.3 Legal Domain related

WordNet domains WordNet domains (Bentivogli et al., 2004) is is a lexical resource created in a semi-automatic way by augmenting WordNet synsets with domain labels. One of the domain labels used is "Law", and therefore a legal domain vocabulary, along with all the information contained in WordNet, can be derived using this resource.

EuroVoc EuroVoc[39] is a European Union multilingual and multidisciplinary thesaurus. It contains keywords, organized in 21 domains and 127 sub-domains (one of them being "Law"), which are used to describe the content of documents in EUR-Lex.

IATE IATE[40] is a terminological database developed by the European Union that contains around 8 million terms in the 24 official languages of the EU. It uses the above mentioned EuroVoc Thesaurus to classify its entries by domain[41].

Discussion

The review undertaken of the resources available to assist in the task of processing temporal expressions and events shows the lack of resources, especially outside the English language. Many of the corpora cited in the literature, moreover, are not available free of charge or are no longer accessible, so they have not been mentioned in this review. Likewise, the resources available in the legal domain are a priori unrelated to temporal information processing, which, together with the fact that there are no corpora of legal documents to test them, greatly hinders the task in this domain.

[38] `https://uvi.colorado.edu/`

[39] `https://eur-lex.europa.eu/browse/eurovoc.html?locale=es`

[40] `https://iate.europa.eu/home`

[41] A more complete list of legal resources can be found in the Lynx Project CKAN: `http://data.lynx-project.eu/dataset`.

2.3 Technologies for Processing Temporal Information

This section reviews the technologies available to handle temporal information. It is structured as follows. Section 2.3.1 covers temporal taggers, this is, the technologies focused on temporal information, usually ascribed to the TimeML standard or to another time-focused annotation scheme. Section 2.3.3 presents Named Entity Recognition systems that partly cover temporal information, considering some temporal information (usually dates) as regular Named Entities, normally not considering types nor normalization, or just covering some of the types. Section 2.3.2 analyzes previous approaches to temporal information in specifically the legal domain. Finally, Section 2.3.4 reviews the software that targets event extraction in a different manner than the TimeML definition of event; thus, semantic role extraction technologies, information retrieval systems, or frame extractor systems are included here.

2.3.1 Temporal Taggers

In this subsection, several temporal taggers in the state of the art are presented. Despite the abundance of works introducing systems for extracting temporal expressions, many of the temporal taggers described in the literature over the past years are no longer available, not maintained, or just work for annotation schemas no longer in use, such as TIMEX2. Thereupon, the present state of the art will introduce the temporal taggers regarding the following selection criteria:

(1) They are operative and widely used, and therefore often cited in literature.

(2) They report good results on corpora from different domains.

(3) They have successfully participated in well-known temporal challenges, such as TempEval-3[42].

(4) They produce TIMEX3 annotations or can be easily adapted to do so.

The temporal taggers can also be classified depending on the approach they follow. Some taggers rely on rules to detect temporal expressions, while others use machine learning techniques, or even both at the same time. On the other hand, normalization is

[42] `https://web.archive.org/web/20200811064918/https://www.cs.york.ac.uk/semeval-2013/task1/`

usually tackled using rules, no matter how the identification is done. Regarding relation and event extraction, both rules and machine learning can be used. The approaches can be therefore the following:

- Rule-based approach: in this case, the temporal tagger uses manually created rules (e.g. regular expressions) that intend to cover all possible paraphrasing on temporal expressions. While this approach tends to perform better than other approaches, especially regarding precision, rules are difficult to scale and are also less flexible than machine-learning approaches.

- Machine-learning-based approach: this approach uses machine-learning techniques, which enables temporal taggers to detect temporal expressions in forms that are not necessarily expected or reflected in existing resources. On the other hand, it requires previous training, preferably done over a large annotated corpora that contain variate temporal expressions.

- Hybrid approach: sometimes, temporal taggers rely both on rules and machine learning techniques. This can be done by using different modules, each of them relying on one of the approaches, or even combining them in different manners, such as for instance increase a manually done rule set by using machine learning techniques.

Besides the kind of approach used by the temporal taggers, the following information is provided (when available) for each of the taggers introduced:

- Supported languages: while most temporal taggers work just for English, some offer additional languages, such as Spanish.

- Approach used: either rule-based, machine-learning-based, or hybrid.

- Covered functionality: which of the tasks among TE identification, TE normalization, relation extraction, or event identification they cover.

- Parametrization options: the possible options offered, such as the style of the text or including or not specific modules of the system.

- Implementation language: the language in which the tool is coded.

- Availability: where it can be found and how it can be used.
- Integration and interoperability with other software: if it can be used along with NLP architecture systems, such as UIMA, GATE (General Architecture for Text Engineering, open source software for text processing (Cunningham et al., 2013))or CoreNLP.
- Dependencies on other resources and required installations: if the temporal tagger requires previous installations before its use.
- Supported output formats besides TIMEX3 (such as JSON or NIF).

All the taggers below have been tested, and an evaluation of their performance on legal texts was published in a journal paper (Navas-Loro et al., 2019a). Table 2.1 shows a comparison of the main aspects of the temporal taggers presented.

T. Taggers	**Approaches**	**Languages**	**Id.**	**Norm.**	**Ev.**	**Rel.**
HeidelTime	Rule-based	ES,EN,DE,+200	✓	✓	-	-
SUTime	Rule-based	ES,EN	✓	✓	-	-
GUTime	Hybrid	EN	✓	✓	✓	✓
CAEVO	Hybrid	EN	✓	✓	✓	✓
ClearTK	Machine-Learning	EN	✓	-	✓	✓
SynTime	Rule-based	EN	✓	-	-	-
TERNIP	Rule-based	EN	✓	✓	-	-
TIPSem	Hybrid	ES,EN	✓	✓	✓	✓
USFD2	Hybrid	EN	*	*	-	*
UWTime	Hybrid	EN	✓	✓	-	-

Table 2.1: Overview of state-of-the-art temporal taggers. The first column indicates the temporal tagger (T. Tagger), the second and the third columns mention the approach followed (if rules or machine learning are used in any way, the software is considered hybrid) and the languages covered, and finally the last columns show the tasks targeted by the software among the four defined by TimeML (Identification, Normalization, Events and Relations). (*) Not all the types are covered.

2.3.1.1 Heideltime

HeidelTime (Strötgen and Gertz, 2012) is a rule-based domain-sensitive temporal tagger created at the Database Systems Research Group at Heidelberg University. It covers more than 200 languages, 13 of them based on hand-crafted resources (namely English, German, Dutch, Vietnamese, Arabic, Spanish, Italian, French, Chinese, Russian, Croatian, Estonian and Portuguese[43]) and the rest of them being automatically created.

Heideltime is capable of processing four different text styles: *News, Narratives, Colloquial* and *Scientific*, the two last ones are only available for English. Regarding the tasks, Heideltime does both TE identification and normalization, having different strategies for each domain, differing for instance in the use of Document Creation Time (DCT) as an anchor for normalization. An example of this is that for news it uses when the document was created as a reference for relative TEs like "*yesterday*", while for other domains, such as narratives, it uses other dates in the text.

Regarding implementation, HeidelTime's Java code, which can be used as a standalone version or via Maven, is available as a public GitHub repository. It can also be integrated into other pipeline environments like the GATE platform (as a plugin) or a UIMA[44] pipeline (as an annotator), and it can be tested online via a demo[45]. It allows both TimeML and XMI as output formats and its architecture is specially designed for being easily extended to more languages or domains.

Since its creation, Heideltime has taken part in several temporal challenges where it has been top-ranked[46], becoming one of the most popular temporal tagging tools due to its language versatility and its ease of use. Nevertheless, to the best of the author's knowledge, it has never been used in the legal domain.

2.3.1.2 SUTime

SUTime (Chang and Manning, 2012) is rule-based annotator for temporal expressions included in the Stanford CoreNLP (Manning et al., 2014). It is built on the TokenRegex tool (Chang and Manning, 2014) (a pattern definition service also part of CoreNLP), and it can both identify and normalize TEs.

A demo[47] and the Java code[48] are available online, and also a GATE plugin and a Python wrapper have been developed[49].

[43] `https://github.com/HeidelTime/heideltime`
[44] `https://uima.apache.org`
[45] `http://heideltime.ifi.uni-heidelberg.de/heideltime/`
[46] `https://github.com/HeidelTime/heideltime/wiki/Evaluation-Results`
[47] `http://nlp.stanford.edu:8080/sutime/process`
[48] `https://github.com/stanfordnlp/CoreNLP/tree/master/src/edu/stanford/nlp/time`
[49] `https://nlp.stanford.edu/software/sutime.shtml#Extensions`

SUTime produces both JSON and TimeML/TIMEX3 tags with new attributes not included in the standard, and sometimes, makes an irregular use of the part of the official specification. For example, set-based TEs (e.g., the expression "*the first weekend of the first quarter*") would produce an alternative value "`2018-Q1 INTERSECT WE-#1`", that is not covered by the standard. By violating the standard, this representation allows SUTime to be more flexible when representing TEs. SUTime presents several related limitations (as analyzed by the authors themselves in Chang and Manning (2012)), offers no domain adaptation, and normalizes with respect to the DCT, if available. SUTime can be used as part of the CoreNLP pipeline as a Named Entity Recognition (NER) system for different languages. Still, the tool works better in English than in other languages (it is unable to process the Spanish version of the expression referring to a weekend previously mentioned, despite having Spanish dedicated rules).

2.3.1.3 TIPSem

TIPSem (Llorens et al., 2010) (Temporal Information Processing based on Semantic information) is a hybrid temporal tagger able to extract temporal information from English and Spanish texts.

It uses both Semantic Role Labeling (Gildea and Jurafsky, 2002) and Conditional Random Field (CRF) (Lafferty et al., 2001) models. Different features are used by CRF recognition models, namely morphological and syntactic considerations at the token level, along with polarity, tense, and aspect information derived from POS tagging and handcrafted rules. Concerning semantic level features, different tools were used to extract information such as each token's role, its governing verb, and lexical semantic information (such as its top class in WordNet (Fellbaum, 1998) or EuroWordNet (Vossen, 1998)) for each token. Similar features are used at the tag level for classification. Also, CRF is used for the normalization task (in combination with rules); finally, the relation extraction features differ depending on the type of relation. TIPSem tackles all the temporal tasks.

The Java code is available online[50], but it requires installation of additional software, such as CRF++[51] (for machine-learning) or TreeTagger (Schmid, 1995) (a language-independent POS tagger), and also optional libraries for improvement of parsing in

[50] `https://github.com/hllorens/otip`

[51] `https://sourceforge.net/projects/crfpp/files/crfpp/`

certain languages (such as Spanish). Also, it must be noted that it is specifically developed for Linux, being its installation in other OS (such as Windows) not encouraged.

2.3.1.4 ClearTK-TimeML

ClearTK-TimeML (Bethard, 2013) is a system that identifies temporal information in English texts using a pipeline of machine-learning models. It is part of ClearTK[52], a framework for developing Machine Learning and NLP in UIMA, and supports TE extraction (and type classification), event extraction, and relation extraction by using specific annotators modeled as BIO[53] token-chunking (for extent/identification of the expressions) or as a multiclass classification task (for types and attribute classification). The normalization task is not covered by ClearkTK-TimeML, being recommended the use of the external TIMEN normalization tool (Llorens et al., 2012). The features used are the ones that proved to be the most successful in previous independent temporal taggers, and are extracted by a morpho-syntactic annotation pipeline with tools not just from the ClearTK framework, but also from others like OpenNLP and Apache, and also a gazetteer.

Written in Java, the ClearTK framework and extensive documentation on the ClearTK-TimeML module can be found online[54]. This tagger relies on external machine-learning tools such as LIBLINEAR (Fan et al., 2008), Mallet (McCallum, 2002) or OpenNLP[55] for techniques like Conditional Random Fields (CRF) (Lafferty et al., 2001) or Support Vector Machines (SVM) (Hearst et al., 1998).

While ClearTK-TimeML does not offer domain-specific adaptions, the pipeline and the parameters can be customized by the user.

2.3.1.5 CAEVO

CAEVO (Chambers et al., 2014) (CAscading EVent Ordering) is a sieve-based architecture, which uses twelve different classifiers pipelined in a cascade way. Despite the default order going from the one with the highest precision to the one with the lowest, the sieves work individually and their position in the pipeline can be freely configured

[52] `http://cleartk.github.io/cleartk/`
[53] Beginning of, Inside of, Outside of a time expression
[54] `https://cleartk.github.io/cleartk/docs/module/cleartk_timeml.html`
[55] `http://opennlp.apache.org/`

(respecting some transitivity constraints), being also possible to add new sieves. These sieves include both rule-based and machine learning-based classifiers, such as WordNet-based and Reichenbach rules (Reichenbach, 1947) based for the first approach, and several CoreNLP MaxEnt classifiers to classify different relations among temporal elements, such as Event-DCT or Event-time.

In contrast to other taggers, CAEVO focuses on the extraction of temporal relations for event ordering, producing *dense* temporal graphs where events and temporal expressions are heavily connected. CAEVO is an expansion of NavyTime (Chambers, 2013) and reuses part of the code of the previously introduced independent temporal tagger ClearTK-TimeML (Bethard, 2013) for part of its sieves. This tagger works just for English texts, covers all the temporal tasks, and has no domain adaptations.

CAEVO is written in Java (working both from command-line or as an API, where it can do batch-processing), and its code and minimal documentation are available online[56].

2.3.1.6 TARSQI

TARSQI (Temporal Awareness and Reasoning Systems for Question Interpretation) (Verhagen et al., 2005) is a hybrid modular system from Brandeis University that covers TEs, temporal relations, and events in English texts.

Each of the modules handles different temporal information. GUTime, based on TempEx, (Mani and Wilson, 2000) was developed at Georgetown University originally for the temporal annotation of news. The approach of GUTime is different from the temporal taggers previously mentioned, as it does not only use rules to find temporal expressions, but it also applies a hybrid approach of rules and machine-learning techniques. The hand-crafted rules serve in GUTime as a basis for temporal annotations that are extended by additional machine-learning ones discovered using the C4.5 algorithm (Quinlan, 1993), i.e. rules to support term disambiguation. Additionally, the module Evita (Events in Text Analyzer) detects events and extracts features such as *aspect* or *tense*. Concerning temporal relations, different modules (Blinker, S2T and the TLink Classifier (Verhagen and Pustejovsky, 2008)) handle the TLINKs, while Slinket (Saurí et al., 2006b) detects subordinated relations between events (SLINKs). Finally, additional modules cover tasks as preprocessing or link merging. The modules of the

[56] `https://github.com/nchambers/caevo`

TARSQI framework can be freely activated or deactivated in order to get just the desired output.

TARSQI is written in Python[57] and is one of the most complete systems for temporal annotation.

2.3.1.7 SynTime

SynTime (Zhong et al., 2017) is a rule-based temporal tagger that proposes a *type-based* approach.

The analysis of corpora performed by the creators showed that most temporal expressions are short, tend to have similar behaviour, and usually contain at least one temporal keyword of a small group of them. Following these ideas, different types of tokens sharing similar syntactic behaviour are defined: *time tokens* (such as 'DURATION', 'YEAR' or 'TIME_ZONE'), *modifiers* (mostly prefixes) and *numerals*. Heuristic rules are built on these types instead of doing it on strings or regular expressions. Since the types are domain independent and the rules work on these types, the system is designed to be domain independent; nevertheless, it must be taken into account that in order to be able to work in different domains, more tokens need to be added for each type. Similarly, in the case of the languages, also the rules should be modified to fit into each language syntax, as they are currently only available for English. SynTime only performs TEs recognition and does not normalize them. For initialization, both tokens and regular expressions over them are collected from the independent temporal tagger SUTime (Chang and Manning, 2012).

SynTime is written in Java and available online[58] as an Eclipse exported project. It uses the Stanford CoreNLP library for POS disambiguation.

2.3.1.8 UWTime

UWTime (Lee et al., 2014) is a temporal tagger from the University of Washington that follows a hybrid approach, using a Combinatory Categorial Grammar (CCG) (Steedman and Baldridge, 2011) parser with hand-crafted rules and learning, along with a hand-engineered lexicon.

[57] `https://github.com/tarsqi/ttk`

[58] `https://github.com/xszhong/syntime`

The idea behind UWTime is using a context-dependent semantic approach for both identification and normalization of temporal expression, with a grammar that translates natural language temporal expressions such as "*2nd Friday of July*" to meaning representation such as '*intersect(nth(2,friday),july)*' (example extracted from Lee et al. (2014)). UWTime just tackles the recognition and normalization of temporal expressions. It uses features such as surrounding tokens and POS, lexical and dependency information, and relies on techniques such as AdaBoost (Freund et al., 1999) for optimization.

UWTime is only available in English with no domain particularities. It can be downloaded online[59], used or be used as an API or as a server. UWTime relies on Stanford CoreNLP software.

2.3.1.9 USFD2

USFD2 (Derczynski and Gaizauskas, 2010) is a temporal tagger from the University of Sheffield that focuses on TEs and relations, based on a previous software named USFD (Hepple et al., 2007).

USFD2 uses a rule-based approach for TEs and both rules and the NLTK's Maximum Entropy classifier for relations (using the features previously established by Mani et al. (2006), a work partially done by some of the creators of TARSQI (Verhagen et al., 2005)). This tagger obtains a good recall with a smaller set of rules when compared with other taggers since they consider specific heuristics for specific tags, such as `DATEs` and `DURATIONSs` as Temporal Expression types, that are the most common. USFD2 only works for English.

The Python code of USFD2 is available online[60], but it must be noted that it is developed for the evaluation of specific datasets, so it must be slightly modified for custom use[61].

2.3.1.10 TERNIP

TERNIP (Temporal Expression Recognition and Normalisation in Python) (Northwood, 2010) is a rule-based Python 2.7 library. It is the result of an MSc in Computer Science with Speech and Language Processing at the University of Sheffield.

[59] `https://bitbucket.org/kentonl/uwtime-standalone`

[60] `https://github.com/leondz/usfd2`, `https://code.google.com/archive/p/usfd2/`

[61] It has been done so for the results later described in this document.

TERNIP is able to both identify and normalize TEs, and its rules can be easily extended following the instructions in its documentation. It only covers English and provides no domain particularities. TERNIP relies on the Natural Language Toolkit library (NLTK) (Loper and Bird, 2002).

It can be used as an API or be integrated as a GATE processing resource, via an XGAPP file (a GATE application file format) available with the code in GitHub[62].

As stated before, besides the temporal taggers cited above, we can also find other temporal taggers in the literature. Nevertheless, they are no longer available, not maintained, or just work for previous annotation schemas. Some examples are UC3M (Vicente-Díez et al., 2010), DANTE (Mazur and Dale, 2009), TEA (Han et al., 2006), JU_CSE (Kolya et al., 2013) or ManTIME (Filannino and Nenadic, 2015).

2.3.2 Temporal Expressions in the legal Domain

The research work by Schilder (2005) already pointed out the importance of temporal information in the legal domain. In this publication, the author extracted events from the United States Code and linked them with temporal information. Schilder proposed to use extracted information for the automated generation of legal narratives or temporal reasoning on legal documents. Also, an analysis of the different types of legal documents and the temporal information that can be found in them was outlined in this work, where Schilder distinguished between dates in transactional documents (this is, documents written by legal practitioners for specific transactions, such as contracts or agreements), constraints in statutes or regulations, and legal narratives in case law. Whereas the first two kinds received dedicated attention, narratives in case law were assimilated to narratives present in the news.

Another approach is that of Isemann et al. (2013), where both NER and temporal processing were applied to extract temporal dependencies from regulations with no narrative structure. The authors faced some of the problems that can also be found in case law, such as temporal taggers confusion between legal references and dates and *episodic* and *generic* statements (about "concrete events" or about "general truths, laws, rules or expectations", respectively).

[62]`https://github.com/cnorthwood/ternip`

Other approaches in the legal domain include temporal expressions and events extraction on legal documents for reasoning (Guda et al., 2011). In some cases, domain knowledge is used (Ramakrishna et al., 2011), and the representation of the temporal information is done in the form of constraint networks. Also, additional efforts focused on evidence and coherence were made (Vlek et al., 2013), using the temporal information but without extracting it from scratch.

2.3.3 NER tools partly covering temporal information

It must also be taken into account that many tools and systems do not consider temporal tagging an NLP task, but just regard dates as a type of Named Entities, similarly to places, currencies, person names, or organizations. This section reviews the most popular of these tools and analyzes to which extent they annotate temporal expressions.

NLTK The well-known Python NLP library NLTK (Loper and Bird, 2002) includes several NE corpora that can be used for training a NER. The annotations will depend on the training, so for having dates annotated it is needed to train with a corpus that includes this type of annotations. Additionally, there is a timex extension[63] that allows to annotate temporal expressions following the standard.

Spacy Another Python state-of-the-art library is Spacy (Honnibal et al., 2020), which uses pretrained pipelines with neural network models for different NLP tasks. Although it does not target temporal tagging, it performs NER, albeit the type of NEs tagged also depends on the model we use. The default one for English (*en_core_web_sm*), for instance, does annotate dates and times (although not together, and considering durations such as "two hours" as time), but not SETs. Also, does not normalize the expressions detected in any way, just marks up them and classifies them as DATE or TIME. Among the NEs this model identifies we find EVENT, but it does refer to events as *an important happening or celebration*, like for instance *Olympic Games*, and not to any type of happening. Nevertheless, models for other languages do not mark any temporal expression as NE.

OpenNLP OpenNLP (Apache Software Foundation, 2014) is an Apache Java project

[63] `https://github.com/nltk/nltk_contrib/blob/master/nltk_contrib/timex.py`

that supports most common NLP tasks, including NE extraction. Different than other approaches, OpenNLP has different models for each task within NE extraction. As can be seen in the available table of models[64], two models are covering temporal information for English, namely dates and time, but again they are not available for other languages and do not provide a normalized value for identified expressions.

AllenNLP AllenNLP (Gardner et al., 2017) is another state-of-the-art Python library thats cover different NLP tasks. Nevertheless, it does not recognize dates in their NER task[65], but does recognize temporal arguments in their Semantic Role Labelling implementation, as will be seen in Section 2.3.4.

CoreNLP On the other hand, CoreNLP (Manning et al., 2014) included their already mentioned temporal tagger SUTime (Section 2.3.1.2) in the Named Entity results of the general NLP library CoreNLP, so temporal expressions can be considered "regular" named entities but at the same time provide more information obtained from the temporal tagger, such as the normalized value or the type of expression according to TimeML.

2.3.4 Event extraction technologies

For the tasks of event extraction, different approaches have been proposed in the literature. Most of them provide their own definition and formalization of the concept of event. Additionally, this section mentions other NLP tasks that, although not explicitly termed "Event Extraction", perform similar processing whose results can be equated to some degree to that of event processing, namely Frame Extraction, Semantic Role Labelling, and Open Information Extraction. This section will review different proposals, from generic to specific legal ones, as well as these related tasks.

2.3.4.1 General approaches

Most of the existing proposals in the literature elaborate their own definition of event, although not always explicitly, and consequently develop strategies in accordance with this concept. In this regard, Hagege and Tannier (2008) observe the difficulty of defining

[64] `http://opennlp.sourceforge.net/models-1.5/`
[65] `demo.allennlp.org`

an event from a concept perspective, so they decide to consider an event any verb (state or action), any deverbal noun, any noun argument of the preposition *during*, or any *time span noun.* For their part, Capet et al. (2008) developed their own ad hoc way to represent events, consisting of some templates with the core of the event and some coordinates (agents, other participants, places, and time). Finally, Chambers et al. (Chambers and Jurafsky, 2008, 2009) consider that verbs sharing coreferring arguments are semantically connected, what they call *narrative coherence*, and they use this information and Semantic Role Labeling to learn new events in an unsupervised way. Some recent proposals also venture to apply neural network techniques to the extraction of events (Feng et al., 2018), but this approach still depends on existing corpora. For a more detailed review, works by Hogenboom et al. (2011) and Xiang and Wang (2019) provide an overview of different event extraction methods.

Existing proposals, therefore, lack a common event definition and often use ad hoc representations for a very specific type of event extraction. Moreover, the annotation formats are too generic and are not adapted to the needs of the legal domain. The ontologies available for representation are also very similar, based on the same information (what, who, and where).

If, on the other hand, we focus on a possible application of event extraction, namely the generation of timelines, we find works like *Linea* (Etiene et al., 2015), a system able to build and navigate timelines from unstructured text, and *TimeLineCurator* (Fulda et al., 2015) a system that is primarily designed to allow journalists to generate temporal stories but can, however, be used to produce a timeline from any free text or URL. Moreover, the timeline generation task has been investigated in other domains, such as journalism (Tannier and Vernier, 2016) or medicine (Jung et al., 2011; Styler et al., 2014).

Related tasks

One of the main difficulties when researching event extraction is that there are many tasks that can be considered equivalent with different names. Some of these tasks are presented below with examples of tools that at least partially cover the event extraction task.

Frame Identification Also called Automatic Semantic Role Labelling[66] or Frame-Semantic Parsing[67], this task involves the use of the frames from FrameNet as a basis for extracting information. One of the most well-known software in this task is SEMAFOR (Das et al., 2014), which is no longer maintained and eventually evolved to Open-SESAME (Swayamdipta et al., 2017), able to find frames in English sentences. Another approach is Framat (Roth and Lapata, 2015), an extension of MATE-tools[68] included in mateplus[69], that uses some of the features previously used in SEMAFOR. The most recent proposal is TakeFive (Alam et al., 2021), which transforms the sentence into a frame-oriented knowledge graph, similarly to other tools like FRED (Gangemi et al., 2017).

Semantic Role Labeling AllenNLP (Gardner et al., 2017) is a deep-learning-based platform able to perform different NLP tasks, such as Semantic Role Labelling[70]. This service "determines the latent predicate-argument structure of a sentence and provides representations that can answer basic questions about sentence meaning, including who did what to whom, etc.". These representations are frames, usually one per verb, with different arguments. IxaPipes (Agerri et al., 2014), a well-known NLP tool for Spanish and other languages also provides an SRL service for Spanish and English[71] based among others in the previously mentioned MATE-tools and PredicateMatrix[72]. The output of the tasks by these two services can be easily assimilated to event extraction.

Open Information Extraction The formerly mentioned AllenNLP (Gardner et al., 2017) also includes an Open Information Extraction service[73] that extracts a list of propositions, i.e. a predicate and its arguments, being these propositions therefore similar to events. For its part, CoreNLP also offers an Open Information Extraction annotator, named OpenIE, that "extracts open-domain relation triples, representing a subject, a relation, and the object of the relation. For example, born-in(Barack Obama,

[66] `https://framenet.icsi.berkeley.edu/fndrupal/ASRL`
[67] `http://www.cs.cmu.edu/~ark/SEMAFOR/`
[68] `https://code.google.com/archive/p/mate-tools/`
[69] `https://github.com/microth/mateplus`
[70] `https://demo.allennlp.org/open-information-extraction`
[71] `https://github.com/newsreader/ixa-pipe-srl`
[72] `http://adimen.si.ehu.es/web/PredicateMatrix`
[73] `https://demo.allennlp.org/open-information-extraction`

Hawaii)"[74]. Similar to the task performed by the AllenNLP service, these triples can, to some extent, be considered events.

2.3.4.2 Approaches in the legal domain

More specifically in the legal domain, existing work also often involves ad hoc definitions of events, ignoring general event annotation schemes such as the ACE 2005 model (ACE2005, 2005). This section will present the different approaches to event extraction in the legal domain divided into two groups, as they relate to different tasks: Legal Information Retrieval (LIR) proposals and Events in Legal Requirements Engineering. Since intrinsically related, also the interpretation of the concept of event used in the work and the representation choice employed, when mentioned in the work, will be referred to.

Legal Information Retrieval

In order to build a case or reason over it, event extraction is a powerful tool for lawyers. In this context, events can be considered as temporally bounded objects that have entities important within the application domain (e.g. persons and organizations) as participants that played a significant role in a case (Lagos et al., 2010). To this end, Lagos et al. (2010) propose an NLP semi-automatic approach to enable the use of entity-related information corresponding to the relations among the key players of a case, extracted in the form of events. They are interested in the topic (*what* happened), the roles (*who* was involved), the location (*where* it happened) and the time (*when* it happened), and consider different types of events, namely role-based events, interaction-based, reference events or cognitive events. Another approach to event extraction for legal case retrieval is that by Maxwell et al. (2009). In this work, any eventuality (event, state, or attribute) is considered to be related to expressions in legal texts, and by its "compositionality" it can be decomposed and composed into great or lesser events. They reviewed 150 events extracted 18 sentences from the Canadian Supreme Court and compared them with automatic extraction using SRL (Semantic Role Labelling) on two cases.

[74] `https://stanfordnlp.github.io/CoreNLP/openie.html`

Some approaches for non-English languages can be found in recent literature. The work by Sierra et al. (2018), for instance, aims to extract events from Mexican legal texts, namely writs of 'amparo' (meaning *protection*). To this end, they look for patterns in documents that help them identify legal events and related information (*who, what, to whom* and *where*), and an analysis was made of the verbs that occur in the texts, as well as the direct objects of each verb. Although no event definition is provided, they consider that events follow a regular pattern involving at least two elements: the action, determined by the main verb, and the date on which the event occurred.

The proposal, which is reported to be still under development, will be evaluated by humans on a corpus of 300 documents. In order to improve information retrieval in Brazilian courts, also a similar work was performed for Portuguese (Bertoldi et al., 2014). In this work, legal events are understood as the cognitive connections that specialists make when they are reading a legal document, and the authors try to recognize possible legal event structures to be described in legal documents. They use semantic frames such as "Lawsuit frame", that has as participants and props 'Type of Action', which indicates the type of lawsuit that was filed against a defendant (administrative, criminal, familiar), 'Author', who is the person that goes to the court with a request, 'Defendant', who is the person that is been suited, and 'Concrete case', which is the legal base that gives the author the right to make a legal request. Nevertheless, this work was reported to be just manual for now, and only ten legal frames have been already identified.

Events in Legal Requirements Engineering

Another possible application of event extraction is the collection of rights or obligations from regulations. This is a different approach because it does not relate to events that actually happen at a precise time with some entities, but to *abstract* events that describe a hypothetical situation that might have some consequences, with some conditions and related constraints.

Kiyavitskaya et al. (2008) aims to automatically extract legal requirements from legal text, namely rights and obligations. Although they do not explicitly define the concept of event nor its components, they elicit "events" as one legal concept, and also date and information. Also another concept elicitation is observed, such as cross-references, actors, or policies. The authors state that they '*'(...) found other terms that we could* *generalize into a common, abstract type, including event, date, and information. Thus, based on the definition section, we derived a list of hyponyms for the basic concepts: actor and policy as well as event, date, and information"*, and used as a corpus the U.S. HIPAA Privacy Rule and the Italian accessibility law.

On the other hand, the Nomos framework (Ingolfo et al., 2014), automatically extracts legal metadata. Although events are not considered explicitly other core concepts related to events (situations, roles) are tackled. Namely, Nomos models are built around 4 core concepts: *roles* (the holder or beneficiary of provisions); *norms* (either duties or rights); *situations* (describing the past, actual or future state of the word); and *associations* (describing how a provisions affects a given situation). A modeling language (Nomos 3) has been implemented. Additionally, events are also targeted in some of the works mentioned in previous sections (Guda et al., 2011; Ramakrishna et al., 2011).

The review above shows that only the second subdomain has actual automatic systems running, while the legislation ones are still ongoing work or are semiautomatic or even manual approaches. It can also be observed that most of the proposals within the legal domain tend to be supported by patterns, using manually crafted rules or semantic role labeling techniques (Kiyavitskaya et al., 2008; Lagos et al., 2010; Maxwell et al., 2009). Other approaches do not search for events specifically but target legal case factors (Wyner and Peters, 2010).

Discussion

This section presented the technologies available for processing temporal information, both in generic texts or in legal documents. Although temporal taggers perform well in texts, there are several specific legal considerations that they do not cover, as will be shown in a subsequent chapter. Similarly, the current state of the art for the Spanish language is not able to handle frequent Spanish temporal expressions. Regarding events in the legal domain, there are no automatic approaches able to target relevant events in judgments.

2.4 Temporal Information Evaluation

This section presents the different evaluation methods proposed in literature, especially in challenges, since they are the main venue where temporal information extraction tools have been presented for competition and performance assessment. While temporal expression evaluation (Section 2.4.1) tends to be always the same, the evaluation of events is usually linked to the representation format or to specific consideration of a challenge (Section 2.4.2).

2.4.1 Evaluation of Temporal Expression Annotation

The most widespread format, the TimeML standard, is usually evaluated using the typical NLP measures Precision, Recall, and F-measure. The aspects usually evaluated in this task are (1) that the extent of the annotation by a tool fits the reference annotation, (2) that the *type* of the expression is correctly classified, and (3) that the normalized value equals the one by the reference annotation and. Optionally, also other attributes of the annotation can be considered, such as modifiers, but this case is not that common.

2.4.2 Evaluation of Event Extraction

Event extraction evaluation is not straightforward. While differences between evaluations of temporal expression extraction are usually limited to the choice among the previously exposed considerations, this is not the case in event extraction evaluation. Usually, evaluation is connected to the interpretation given to what an event is. If, for example, we compare the TimeML event concept, approaching it from a purely temporal point of view and marking it as temporal entities with attributes and linking it only to other temporal expressions, with the proposal by Ji et al. (2009), which focuses on the event's arguments detection, focusing its evaluation on Named Entities and the temporal ordering, it becomes clear that the evaluation of both approaches cannot be the same. Furthermore, the task addressed by Ji et al. (2009) is slightly different, as it involves handling cross-document extraction, a context in which common Named Entities help to link related events, which are also more difficult to order than in a single document, which also tends to have narratives that tend to be chronologically ordered.

On the other hand, challenges in the domain have also influenced how events are annotated and evaluated. In this regard, SemEval is one of the best-known NLP challenges, and for many years it has had tasks related to temporal information, which became the reference venue for different proposals of temporal taggers to measure and compete with each other. However, this competition, called TempEval, did not always take place in the same form, and over the years the way of evaluating also evolved. The first editions (Pustejovsky et al., 2009; UzZaman et al., 2013), for example, used the TimeML standard for the annotation of the benchmark datasets, and for the evaluation used Precision, Recall, and F-measure to assess both the correct extent of the annotation and the accuracy of the tool in assigning the event type (out of the nine classes defined in the TimeML guidelines). Optionally, this evaluation could also include other non-primary attributes of the events in TimeML, such as time, aspect, polarity, and modality. Additionally, later another task in the challenge involved the extraction of temporal relations, which could be considered complementary to event extraction, but not part of its formalization per se. However, in the 2015 edition of SemEval (Llorens et al., 2015) the evaluation shifted to temporal question-answering, thus prioritizing temporal comprehension over simple annotation in the text, where simply differing in the extent of the annotated event was considered an error. On the other hand, the ACE 2005 evaluation scheme (ACE2005, 2005) proposes the ACE VDR (Event Detection and Recognition) value, a metric developed taking into account both the extent of the event annotation and the event arguments and their attributes *value* and *modality*. Sadly, in the last editions of SemEval, just the medical domain temporal task Clinical TempEval[75] persisted.

Meanwhile, other proposals involve several levels of evaluation. This is the case of the BioNLP'09 shared task on event extraction[76], where different tasks addressed different levels of detail of event extraction, from the mere identification of the so-called 'event core' (i.e. its trigger, type, and main argument) to the surrounding entities (such as the place of the happening) and its factuality (e.g. whether it actually occurred, or is negated or presented as a simple possibility).

[75] `https://competitions.codalab.org/competitions/15621`

[76] `https://web.archive.org/web/20191023015953/http://www.nactem.ac.uk:80/tsujii/GENIA/SharedTask/`

Besides this evaluation of the form in which events are annotated, also how to decide which events to annotate and which ones should not be considered. Regarding the legal domain, a judgment may contain thousands of events in the broadest sense of the word, but not all of them are equally relevant from the legal point of view. For this reason, some efforts have previously been made to discern significant events (Chasin, 2010). However, no specific evaluation metrics on this aspect have been found in the review of the literature. Only when these events were subsequently used for an alternative task (e.g. summarization), some evaluation was done using metrics of the final format (Prasojo et al., 2018).

Discussion

Despite being intrinsically linked, the evaluation of the different tasks framed in temporal information extraction is not at all equivalent. While in the case of temporal expressions it is widely accepted to evaluate with respect to very specific and closed aspects, always with the same standard and its respective guidelines as a reference, this is not so in the case of event evaluation. On the contrary, there are various interpretations and formalisations of what an event is, so how to evaluate them is not a consensual task. Each of the different conceptions of event implies different representations and attributes to consider, thus necessitating different metrics to measure correction and even different levels of evaluation.

Summary

This chapter has reviewed the state of the art of four different fields related to temporal information.

First, the different options available for representing temporal information were presented, with a special focus on the most widely used standard, TimeML. Besides this, other available annotation schemes have been briefly presented, as well as ontologies and some legal domain representation options. We can conclude that while TimeML is the established standard for temporal expression annotation, events are usually tackled from an ad hoc perspective and there are plenty of options available, lacking a single consensual way to represent them.

Second, the available resources related to temporal information were examined; both corpora and other resources were presented, either focused on the legal domain or of generic use.

Then the different technologies available for processing temporal information were analyzed. Several state-of-the-art temporal taggers were presented, but also tools in the legal domain, NER options partly covering temporal information and both generic and legal event extractors.

Finally, different evaluation options for temporal expressions and events were briefly reviewed.

Chapter 3

Particularities and Challenges

Legal documents differ in structure and writing style from other documents. According to Dell'Orletta et al. (2012), legal documents are characterized by domain-specific linguistic phenomena, namely raw text features, lexical features, morpho-syntactic and syntactic features. Among them are the length of sentences (being legal ones larger and more complex on average), differences in the distribution of part-of-speech (legal documents tend to have more prepositions, supporting the claim of their grammar being more complex), and syntactic features (legal texts have deeper dependency parsing trees, but lower arity of verbal predicates).

In order to extend this analysis to time-related particularities, two different approaches were used. The first one was an exhaustive analysis of legal documents, the temporal expressions in them, and the compatibility with the TimeML standard. This analysis was completed with an evaluation of ten state-of-the-art temporal taggers on a corpus of court decisions (that will be introduced in Section 5.1.3) where some of the particularities found became evident. The second approach, focused on industry, consisted of asking experts in the legal domain about their specific needs regarding time expressions. To this aim, they were asked to fill in a questionnaire and to later check a proposal to deal with their needs.

Section 3.1 presents an analysis of temporal expressions present in judgments, while Section 3.2 does so with events. Finally, Section 3.3 addresses various shortcomings detected in information processing in general, in terms of the standard, the current implementations, and the needs to be covered.

3.1 Analysis of judgments

A first analysis of judgments[77] revealed several particularities, detailed below. This analysis was performed on a corpus of 30 court decisions from three different courts in English, namely the European Court of Human Rights (ECHR), the European Court of Justice (ECJ), and the United States Supreme Court (USSC). The corpus, along with its building methodology and analytics, is presented in detail in Section 5.1.

3.1.1 Temporal Dimensions

In a judgment, several temporal dimensions can be found. Each event and therefore each temporal expression can be attributed to different temporal threads. An analysis of legal judgments from different courts suggested distinguishing three main temporal dimensions in judgments, namely:

- *Temporal dimension of the legal process*, where events and time expressions related to the legal process, such as previous legal hearings or decisions of the case, would belong. These events are based on some standard rules and new events that are part of standard court proceedings, gradually building the legal proceeding. When dealing with European, federal, or international courts, there can be also a distinction here among local or national courts (*background procedural*) and the court issuing the judgment that is being processed (*procedural*). Dates of decisions or applications would belong to this temporal dimension.

- *Temporal dimension of the case*, where events and time expressions related to facts under judgment, such as murders or robberies, would be placed. These events are the ones that unleash the legal procedure.

- *Temporal dimension of the legal context*, with temporal expressions unrelated to the case part of the applicable legal context, such as dates of laws affecting the legal process or related jurisdiction. This information can be relevant in order to check the law in force when the different events in the judgment happened.

[77]This work is partially the result of a collaboration with Erwin Filtz, Sabrina Kirrane, and Axel Polleres (along with Víctor Rodríguez-Doncel, co-advisor) during a research stay in the Wirtschaftsuniversität Wien. Therefore, part of this analysis has already been published in a high-impact journal (Navas-Loro et al., 2019a).

There is also other temporal information appearing in judgments (*background information*), such as the birth date of the defendant, that can be relevant or not to the case and therefore be included or not in one of these temporal dimensions. If for instance, the date of birth implied that the defendant was a minor at the moment when the facts under judgment happened, or the date of birth implied for instance that the person obtained some nationality, it would be relevant to the case. An example of each of these dimensions, appearing also in Navas-Loro et al. (2019a), are the following excerpts from the case *Sophie Mukarubega v Préfet de police and Préfet de la Seine-Saint-Denis* (ECLI:EU:C:2014:2336):

> "By a decision of 21 March 2011, adopted after hearing the person concerned, the Director General of the Office francais de protection des refugies et apatrides (OFPRA) (Office for the protection of refugees and stateless persons) rejected her application for asylum. (...)"

This temporal information belongs to the *temporal dimension of the legal process*, since it is a rejection of asylum.

> "Ms Mukarubega, who was born on 12 March 1986 and is of Rwandan nationality, entered France on 10 September 2009 in possession of a passport bearing a visa. (...)"

The entrance of France is part of the *temporal dimension of the case*, while the date of birth would be out of the temporal dimensions of interest.

> "(...) This request for a preliminary ruling concerns the interpretation of Article 6 of Directive 2008/115/EC of the European Parliament and of the Council of **16 December 2008** (...)"

Finally, this date in bold is related to a directive of the European Union, not to a legal event of the case or the facts under judgment, so it would belong to the *temporal dimension of the legal context*.

Additionally, sometimes the belongingness to one or another temporal dimension can be ambiguous, or even overlapping. Since most legal systems are composed by hierarchies of courts where cases might go from court to court and eventually back to a previous one, sometimes it is not easy to place an event in the *temporal dimension of the legal process* or in the *temporal dimension of the case*. It is considered (Navas-Loro et al., 2019a) that *temporal dimensions of the case* just contains facts of the case, while subsequent revisions, decisions, and case remands do not affect or alter in any way these original events, but just the further legal proceeding, and would therefore be part of the *temporal dimension of the legal process*. Figure 3.1 shows a timeline[78] with the different temporal dimensions[79] of a real case of the European Court of Justice, namely a request for a preliminary ruling. A summary of the case[80] for better understanding can be found below.

> The request has been made in proceedings between Mr Sergejs Buivids and the Datu valsts inspekcija (National Data Protection Agency, Latvia) concerning an action seeking a declaration as to the illegality of a decision of that authority, according to which Mr Buivids infringed national law by publishing a video, filmed by him, on the internet site www.youtube.com of the statement which he made in the context of administrative proceedings involving the imposition of a penalty in a station of the Latvian national police.

In this example, the third event in the timeline (*"The National Data Protection Agency requested Mr Buivids to remove that video from the internet site www.youtube.com and from other websites"*) is considered as part of the *temporal dimension of the case*, although it might be considered part of the *temporal dimension of the legal process*, since it is a formal request produced by a decision of a National Agency. Nevertheless, is this precise event what is being judged in the Latvian Courts and in the European Court of Justice, and not the fact of uploading the video *per se*.

The importance of this categorization is not only conceptual but also empirical. The time expressions in the different temporal dimensions should be normalized always considering the dimension they belong to. Some examples of this are included in the following subsection.

[78]Since some of the events are not dated, they were put in the timeline in chronological order, so the placement might not be exact.

[79]No time expressions of the *temporal dimension of the legal process* were added to this timeline for the sake of readability.

[80]This summary was gently provided by Cristiana Santos.

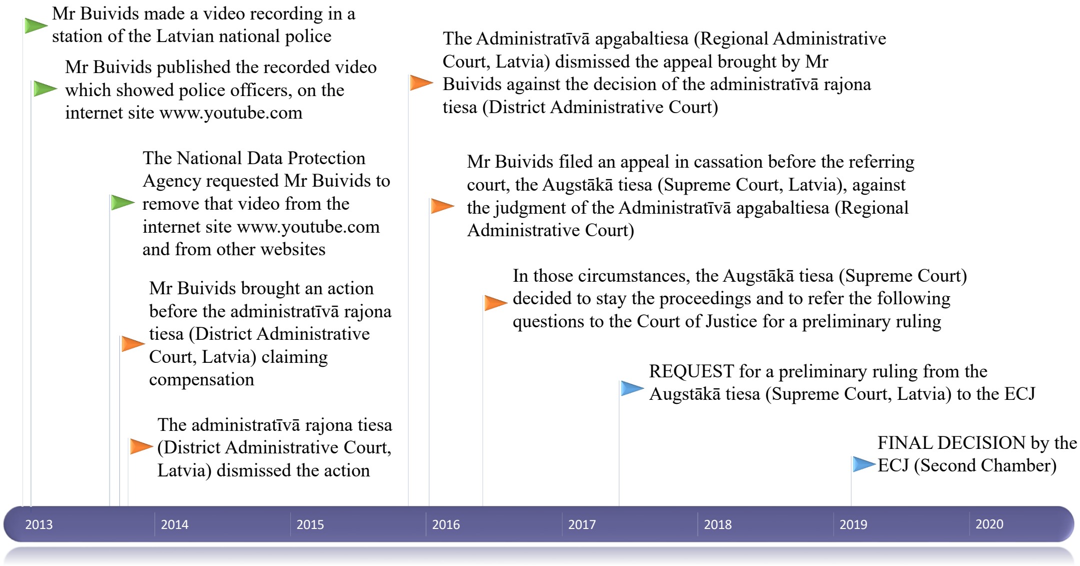


Figure 3.1: Timeline of the case 345-17 of the European Court of Justice. Blue landmarks correspond to the *temporal dimension of the legal process*, namely to the *procedural* events carried out by the last instance court (in this case, the European Court of Justice). Orange events also correspond to the *temporal dimension of the legal process*, but have been catalogued as *background procedural* (this is, events happening in national Latvian courts). Finally, green landmarks represent the facts under judgment, in the *temporal dimension of the case*, that are the events that unleashed the whole legal procedure.

3.1.2 Misleading Expressions in Legal Documents

Legal references usually follow text patterns that include numbers and symbols (such as '/') that tend to be confused with time expressions. These misleading expressions might actually contain time expressions (such as the year a law entered into force or was published, or a court case was decided), or be just misleading expressions (see Table 3.1 for examples of both). Despite some of these expressions are actually referring to points in time in the legal context and might be therefore tagged, doing so might lead to normalization errors in the surrounding temporal expressions. This problem was analyzed and partially tackled in one of the publications of this thesis (Navas-Loro, 2017).

The following excerpt, extracted from the ECJ case C-520/06 (ECLI:EU:C:2009:18), is an example of this:

> (...) In the light of the foregoing, the answer to the first question referred in Case C-520/06 is that Article 7(1) of Directive **2003**/88 must be interpreted as not precluding national legislation or practices according to which a worker on sick leave is not entitled to take paid annual leave during that sick leave.
>
> *The right to paid annual leave in the event of sick leave which lasts for the whole or part of the leave year, where the incapacity for work persists beyond* ***the end of that year*** *and/or of a carry-over period laid down by national law* (...)

Since some temporal taggers consider that '2003' (part of a legal reference, in bold) is a year and therefore tag it as a temporal expression, depending on the normalization strategy used, the value of the next temporal expressions might be affected and badly normalized with regard to it. In this case, 'the end of that year' might be normalized as *2003*, which is not correct. Nevertheless, taking into account the temporal dimensions, while the first expression in bold would belong to the *temporal dimension of the legal context*, the second one would be normalized with regard to a date of the *temporal dimension of the case.*

Similary, in the following excerpt, extracted from *Howel v. Howel*[81] (judgment from the US Supreme Court and part of the TempCourt corpus, introduced in Section 5.1):

> (...) amounts deducted from that pay"as a result of a waiver . . . required by law in order to receive" disability benefits, §**1408**(a)(4)(B). The divorce decree of petitioner John Howell and respondent Sandra Howell awarded Sandra 50% of John's future Air Force retirement pay, which she began to receive when John retired **the following year**. (...)

While some temporal taggers consider that '1408' (a legal reference, in bold) looks like a year and therefore tag it as a temporal expression (false positive), depending on the normalization strategy used, the value of the next temporal expression might be affected and badly normalized with regard to it. In this case, 'the following year' might be normalized as *1409*, what is obviously not correct. This case stresses the need for distinguishing legal references, whether they contain temporal expressions dates or

[81] `https://www.supremecourt.gov/opinions/16pdf/15-1031_hejm.pdf`

not. Legal references should be detected in a pre-processing step, and hidden somehow to the temporal tagger task, or be filtered after the identification task but before the normalization. A possible way to tackle them is the one implemented in *lawORdate* (Navas-Loro, 2017), a software that processes a text looking for legal references and replacing their mentions for innocuous expressions that will be presented in Section 6.2.

Source	**Example**	**Description**
ECHR	no. 7334/13, 127 - 128, ECHR **2016**	Reference to another case
ECHR	Timoshin v. Russia (**dec.**)	Reference to a decision (dec.), often confused with the month of December
ECJ	OJ **2008** L 348 p. 98	Reference to official journal of the EU
ECJ	Directive **2008**/115/EC	Reference to a directive published in 2008
USSC	See Va. Code Ann. §53.1-165.1 (**2013**)	Law reference
USSC	[...] 772 F. 3d **1328**, **1333** (CA10 **2014**)	Precedent case reference

Table 3.1: Examples of misleading legal references in legal texts. All the examples were taken from the TempCourt corpus (to be introduced in Section 5.1), and the sources are the European Court of Human Rights (ECHR), the European Court of Justive (ECJ) and the United States Supreme Court of Justice (USSC). This table is taken from Navas-Loro et al. (2019a).

3.1.3 Structure of Court Decisions

While court decisions as a type of document share common characteristics, such as the narrative style in part of its text, their structure tends to differ hugely from one court to another. Table 3.2 illustrates the structure of documents from three different courts in English (ECHR, ECJ, and USSC). Each of the sections hints the different types of events and TEs to be found. The different structures are detailed hereunder.

European Courts (ECHR, ECJ) Both courts follow a similar structure, that is outlined herebelow following the naming provided in Table 3.2). The structure of ECJ is followed, mentioning the differences present in ECHR documents if any:

Section	ECJ	ECHR	USSC
A	Involved parties	Involved parties	(Syllabus)
B	Case summary	Procedure	Main Opinion
C	Legal framework	Circumstances of the case	Concurring and dissenting opinions
D	Circumstances of the case	Legal framework	
E	Court assessment	Court assessment	
F	Judgment	Judgment	

Table 3.2: Structure of ECJ, ECHR and USSC decisions. Modified from Navas-Loro et al. (2019a)

A. Involved Parties. Both European decisions start with the different participants involved in the procedure. This section is not explicitly named in the document; this is, it is not the title of a section, but an intrinsic part of the introduction. In the case of ECHR documents, it tends to include just the members of the Committee to evaluate the case. On the other hand, ECJ includes the names and roles of all the participants, such as the Advocate General, and other Agents involved. Similarly, the names of the applicant and the defendant are introduced, but not explicitly naming their roles, but following a standard formulation [APPLICANT] v [DEFENDANT].

B. Case Summary/Procedure. ECJ court decisions include a short description (usually 2 paragraphs) that summarize the case in legal terms (e.g., the Articles of the European law needing interpretation) and also a short overview on the case, mentioning the main participants of the facts being judged and stating the nomenclature of these parties in the remaining of the document, as done in the following excerpt from Case C-34/13 of the ECJ:

> *"The request has been made in two sets of proceedings between the Minister voor Immigratie, Integratie en Asiel (Minister for Immigration, Integration and Asylum) ('**the Minister**'), on the one hand, and, respectively, Ms S. and Ms G., third-country nationals and family members of a European Union citizen of Netherlands nationality, on the other, concerning the **Minister**'s refusal to grant them a certificate of lawful residence as a family member of a Union citizen in the Netherlands."*

On the other hand, ECHR documents include a section of FACTS or FACTS AND PROCEDURE where also a short description of the case is presented.

C. Legal Framework. In the case of ECJ court decisions, the following section, named “Legal Context”, includes (i) The European Law relevant to the matter, and (II) The National Law to be considered. The relevant paragraphs of the law are cited explicitly; there are no mentions to the case under judgment, but it is just a recompilation of the law related to it. Moreover, in the case of ECHR, this Legal Framework section is not the third (C) but the fourth (D), and does not include the laws referred to, but just cites them with regard to the different facts of the procedure. The Legal Framework sometimes refers also to external documents of the case for checking part of the law in the FACTS part of the document.

D. Circumstances of the case. ECJ court decisions succinctly present in this section the facts under judgment, along with the previous legal procedural events, such as previous resolutions from national courts. This section might be subdivided into subsections such as “X situation”, referring to the circumstances surrounding the different parties. ECHR, on the contrary, includes the main facts of the case in the FACTS (and PROCEDURE) part of the document, sometimes with additional subsections for dividing them (such as “Circumstances of the Case”).

E. Court Assessment. This section contains the matching between the law and the case in the documents from both courts.

F. Judgment. The final part of the document, that states the final decision. In ECJ documents the judgment usually takes several paragraphs, while in ECHR it is just one sentence with the final resolution.

Although this structure is often followed, it must be noted that some additional sections can be found in these documents, such as a “COMPLAINTS” section between “THE FACTS” and “THE LAW” sections in certain cases from the ECHR. Similarly, some sections with no relevant content to temporal information are ignored, such as the signatures at the end of the documents from the ECJ.

Non-European court (USSC) The documents from the United States Supreme Court present a different structure than the European ones previously introduced. The main information derived about structure is outlined below:

1. First of all, it is not a single structured text, but a collection of texts concatenated. Each of them has a title and a header.

2. Each independent text is not clearly structured into named sections, but into numbered sections.

3. Also, some of the documents analyzed in TempCourt include a section called "syllabus", written for sake of understanding, but it is not a part of the decision.

4. Since there are different texts included in each document, it is common to find pieces of text repeated throughout the document.

A. Syllabus. This section is not part of the judgment itself (as stated by a disclaimer in the document), it is added as a summary of the case to facilitate its understanding. The syllabus could be comparable to some extent to the *Circumstances of the case* section of the European decisions explained earlier in this chapter.

B. Main Opinion. This section shows the Opinion of the Court, including both facts and references to both laws and previous cases throughout the text.

C. Concurring and dissenting opinions. Additionally to the main opinion of the court, also the opinion(s) of other judges can appear in the document. An opinion can be *concurring* if the judge(s) involved agree with the main opinion but fund the decision on different grounds. On the other hand, *dissenting* opinions show the discomfort from some judge(s) about both the decision and the grounds offered in the main opinion. Several opinions can appear in the same document.

Despite the obvious differences between European courts and the US Court, the underlying presence of a structure can always be assessed. The information found in documents from both sources is similar, just differently organized. Knowing this source in advance is crucial for adequate parsing, especially when considering the temporal dimensions described in Section 3.1.1. For instance, dates appearing in the Legal Framework will belong to *the temporal dimension of the legal context.* On the other hand, events and dates appearing in the "Circumstances of the Case" section of the European documents will be, for instance, probably part of *the temporal dimension of the case*; nevertheless, in the case of purely legal events like "appeal", they can also be on the other hand considered *background procedural events*, and therefore be part of the *the temporal dimension of the legal process.* Regrettably, making this distinction is particularly difficult in the case of the documents from the USSC, since all the temporal dimensions are mixed in the text.

3.2 Events in the legal domain

Regarding events in the legal domain, the first analysis on them was performed in collaboration with Cristiana Santos (Navas-Loro and Santos, 2018), being the content of this section extracted from the derived publication. This analysis, similarly to the previous one, can be divided into two parts. First, Section 3.2.1 presents some general observations on the concept of legal events. Afterwards, Section 3.2.2 outlines more specific considerations derived from the first annotation of a corpus of judgments.

3.2.1 General Observations

Legal events (and the annotations thereof) may vary according to different criteria contingent on the legal realm. Possible differential criteria in a general perspective are presented below:

- Multijurisdictionality and multilingualism: legal events vary according to the common law or civil law jurisdictions, as well as to the languages in which they are expressed or into which they are translated.
- Document dependency: the qualification of legal facts may vary according to the heterogeneity of the document in which they are inscribed: e.g. it is not the same a contract and a court decision, and among court decisions, we find landmark cases and commonplace cases, as well as different legislation (primary, secondary, domestic, European, international). In addition, both the jurisdiction and the underlying domain of the document (civil, criminal environmental, taxes, business, etc.) are pertinent when determining the relevance of an event. For instance, comprising the domestic judicial hierarchy and their procedural rules,

and the procedural rules of the higher Courts, e.g. request for a preliminary ruling, hearings, submitting observations, opinions, citing case-law, etc.

- Level of abstraction: legal events can be considered from a casuistic analysis (a specific argument of a case) to a general consideration of the facts of a case. For example, the same event (e.g., someone declares something) can be seen as a declaration (a reporting event), a procedural event (specifically, part of the timeline of the proceedings) or as an abstract legal event (an event in the document level); also, what is being reported could suppose events on their own with some confident score.

- Agents and role: the consideration of legal events can vary according to an agent, which is a participant in some juridical relationship, e.g., the applicant (a victim vs perpetrator), or a reputed judge, a notary, a legal scholar. The role of the court is also influenced by their level of authority (first, second national instances).

- Temporal, contextual and spatial features: this quadrant (time, context and space) can be illustrated by chronological events within court proceedings, such as the submission of an application on a certain date in the applicant's national Court; pleadings; ulterior appeals to a different located court; judgment delivery by the ECJ (located in a different state) –as the ultimate decision that ends all proceedings, etc.

- Scenario or application-based: annotation of legal events might vary according to the sought application, purpose or scenario. If one considers predicting judicial decisions, events referring to case facts will be mostly regarded (Aletras et al., 2016). If, however, the devised application aims to detect arguments for legal argumentation, then the party's claims will be the target annotation (Lippi et al., 2018). The same holds for considering the most cited case-law to consider the authoritative and relevant ones (van Opijnen, 2016).

These criteria give a hint on the different aspects to take into account when sketching the first definition of an event in the legal domain. Regarding their annotation, the next section shows some empirical insights.

3.2.2 Lessons learned from the case study

This section presents the lessons learned while annotating the events in a corpus of 10 European Court of Justice (ECJ) decisions:

- Events can be found in several forms, such as verbs (declare), nouns (appeal) and nominal phrases (the facts of the case).
- As usual while processing texts in the legal domain, legal terminology, characterized by synonymy, ambiguity, vagueness, polysemy – suppose an extra challenge. This thesis has captured the following variations:

  a. Conventional terms change their meaning in the judicial decision-making; this is the case, for instance, of the expressions "lodging an application *before the Court*", or "criminal proceedings were instituted *against the applicant*", where the terms *before* and *against* do not have the conventional meaning;

  b. Terms can have several interpretations. For instance, two verbs (*submit, argue*) are indicative expressions of an argument. However, the verb "submit" can also refer to submitting written observations or pleadings, which consist of procedural documents lodged before the Court.

- In the preamble different types of *event-aware information* can be found regarding to:

  a. *Identity-related event* on information related to some of the participants involved in the main proceedings:

     i. The referring court, and ECJ (and its internal composition);

     ii. The litigant parties: applicant/defendant

     iii. Agents (States, European Commission, etc);

  b. *Location-event* and *date-related event*: it is possible to identify both the date of the request and the place of the national referring court, e.g., *"10 September 2014, Request for a preliminary ruling* from *the Krajsky Sud v Presove (Slovakia)"*.

  c. *Domain-related event* of the judgment: the initial summary of the judgment indicates for the domain at stance, e.g., consumer protection (or others, illegal migration, genetic modified food, etc.)

- The background of the case is the most interesting part to annotate, as it includes the relevant events, arguments and facts of the parties, e.g. *"On 26 February 2009, Mrs Kusionova* <u>*concluded*</u> *a consumer credit agreement with SMART Capital for an amount of EUR 10 000"*, Case 34/13, paragraph 25.

- Events can be subsumed to decisions of the national courts (first-instance and second-instance courts that refer to the ECJ). Expressions on the text mentioning "Regional Court", "District Court", "national court" illustrate what are the juristic positions of the former courts according to a legal problem.

- Interpretative issues were considered when analysing the different versions alleged by each of the parties in the dispute concerning the same event, e.g. the claimant alleges there was an illegal use of goods and no smuggling. Each of the involved parties claims that the other one is at fault, which consists of an interpretative indicator of the same event.

- Negation and Factuality: it must be noted that some events can be negated or not actual facts, but "possibilities, intentions or preferences" (Navas-Loro et al., 2019b).

- As for event-related relations, we observed that two-way (bi-directional) relationships can be found in the same judgment engaging both parties, e.g. actions "submitted by", "brought by"; or "the facts of the case, as submitted by the parties'; or "observations submitted by the Government and the observations in reply submitted by the applicant".

- Legal related events can be identified at different (internal) structures of the documents, e.g. at a paragraph level of a court decision, or in the summary, or in the conclusion thereof.

This initial analysis, based on a small corpus of ECJ annotated judgments, permitted to outline an initial scope of legal events. The qualification of events in the legal domain seeks to be wide-ranging in scope and facilitating the detection and extraction thereof, regardless of their applicable domain (criminal, civil), but customizable/modular for instantiation. Still, one of our main concerns was *relevance*. A judgment can contain thousands of events in the broadest sense of the word, but not all of them are relevant from the legal point of view. How could relevance be measured? By the appearance of this event in other judgments from the same court? From the same legal field? Is it related to the agents intervening in the event? Is it possible to extract a common structure of most legal procedures in a court? A jurisdiction, a country? Further work done in this line of research, trying to answer some of these questions, is detailed in the third part of this dissertation.

3.3 Real world needs

Besides the gaps already mentioned, regarding industry needs a questionnaire was distributed among Lynx partners[82] (namely, the legal company OpenLaws and the law firm Cuatrecasas) in order to evaluate their needs in the legal domain. This document is provided as Annex 1. Additionally, a collaboration with the CENDOJ (Centro de Documentación Judicial, Center of Judicial Documentation[83]) allowed us to detect some important gaps. A summary of the lessons learnt about lacks and possible improvements of the TimeML standard concerning real needs are outlined below:

- The implementation of intervals is not straightforward. This can be done in the current version of the standard, but via the DURATION type, and this option is not used by the temporal taggers.

- There are common expressions that cannot be represented using the standard, such as "every other day". Also, other expressions cannot be properly represented with the current constraints (e.g. "every night", since *night* for the standard is not a granularity but a part of the day).

- In the case of abstract temporal expressions there is no way to keep the existing information. For instance, if a contract reads "within 6 days from the day of the signature", the normalization depends on a date that is probably not provided with the text. It would therefore be useful to be able to leave the value as a function of a date to be known in the future (e.g. SIGNATURE_DATE + 6D). Although TimeML has similar specific functionInDocument attributes such as

[82] European project in which I participated during this thesis `https://lynx-project.eu/`
[83] `https://www.poderjudicial.es/search/indexAN.jsp`

PUBLICATION_DATE, we are not aware of the option to use them as variables in the value of temporal expressions.

- Event attributes are limited to the linguistic and temporal point of view. For example, there is no way to add an actor to an event.

- There is no way to express nested, fuzzy or composed expressions. For instance, in the sentence "I will attend on Thursday or Friday", the answer to "When will I attend?" should be "Thursday or Friday", but following the standard, we should tag each day separately and there is no way to express the intended meaning.

- The standard is very English focused. It does not allow other calendars, such as the Japanese one. Also, the parts of the day it allows to represent do not have correspondence with other languages.

- It does not facilitate handling context-dependant temporal expressions (these expressions will be discussed in Section 6.3).

Concerning the way that state-of-the-art temporal taggers implement TimeML annotations, also some shortcomings have been detected.

- Most optional attributes are never used, even if they are useful (e.g., modifiers) because they do not count in the evaluation of related challenges.

- TimeML EVENT annotation is less common than temporal expression annotation. Temporal taggers are often just used for temporal expression annotation, while the event extraction task is done otherwise.

- Most temporal taggers are not able to detect compound durations, since it is not usual to find them.

- Since it is the way they are usually evaluated, most temporal taggers focus on covering expressions that can appear in news.

We additionally find some challenges in Spanish temporal tagging that are not covered by current temporal taggers. For instance, polysemous expressions such as "mañana", which has several meanings in Spanish (both time-related and unrelated), are usually wrongly annotated. Also, Latin American temporal expressions, such as

“Cinco para las diez” (“Five to ten”), are not covered. These cases will be discussed in more detail in Chapter 6.

Summary

This chapter presented some of the particularities and challenges detected in the legal domain and in the temporal information processing field.

First, the different difficulties that can be found in legal decisions, such as misleading expressions or the existence of different temporal dimensions, were analyzed. Afterwards, some observations and lessons learned from a first experience annotating events in legal judgments were detailed. Finally, some real world problems in temporal annotation, both related to the temporal taggers and to the TimeML standard, were outlined.

Chapter 4

Goals and contributions

This chapter presents the main goals of the thesis (Section 4.1) and the main contributions done to achieve them (Section 4.2). In addition, also the assumptions taken (Section 4.3), the hypotheses (Section 4.4) and the restrictions considered (Section 4.5) are detailed. Finally, the research followed methodology (Section 4.6) and the evaluation methodology (Section 4.7) are presented.

4.1 Goals

The main objective of this thesis is **to improve the temporal information extraction and representation in the legal domain** by creating dedicated tools and new semantic resources that facilitate their visualization and use in further processing tasks.

O1. To **analyze the particularities and needs of the legal domain users with regard to temporal information** (temporal expressions and events). The objective is to tackle them in a way that helps both legal practitioners to improve their work and layman to understand long and complex documents.

O2. To **help processing temporal information in legal texts in Spanish and English**. Also, the lack of resources and technologies in the Spanish language in general will be targeted, and the objective is to encourage the use of temporal information in subsequent tasks.

O3. To **be able to transform textual judgments into series of events**. This event-based representation would allow expressing judgments in intuitive and easy to understand ways, such as timelines or event-based knowledge graphs, or to enhance further semantic tasks such as summarization, pattern recognition or event-based search.

In order to achieve these goals, the following limitations detected in the State of the Art had to be addressed:

L1. The last version of the standard commonly used for temporal annotation is back from 2006 and is not adapted to the legal domain. On the other hand, there exist many ontologies and data models to represent temporal information, but none of them is intended for NLP and able to support the storage of relevant information of the annotation task.

L2. Current state-of-the-art temporal taggers for Spanish perform good results in competitions, but underperform when dealing with common Spanish expressions that are not present in the corpora used in these challenges.

L3. The current temporal taggers do not perform well on legal texts, since they have not been designed nor trained for this kind of documents.

L4. Lack of corpora including temporal information processing, especially in the legal domain, as has been presented in the state of the art.

L5. Existence of many models for event representation, but not one single standard or unifying approach (as discussed in the state of the art of Chapter 2).

4.2 Contributions

The main contributions of the thesis are outlined below. In order to centralize their access, a webpage[84] has been created. Fig. 4.1 relates these contributions to the hypotheses, constraints and assumptions.

C0. Analysis of the temporal information in the legal domain. An analysis of the particularities and challenges in the legal domain with regard to temporal information processing. This analysis was presented in Chapter 3.

[84] `https://mnavasloro.github.io/PhDContributions/`

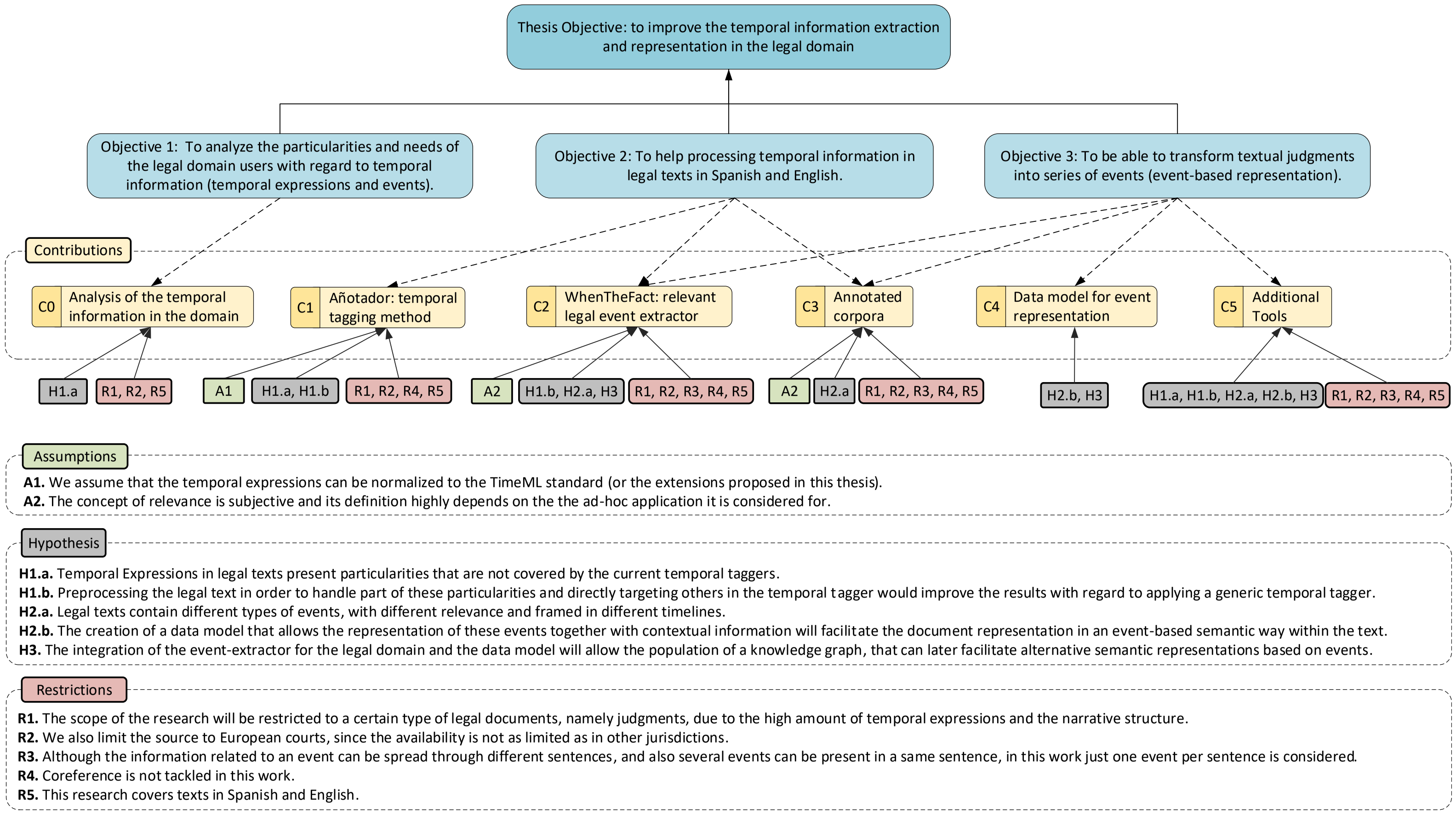


Figure 4.1: Conceptualization of the thesis. Mapping of the objectives, contributions, assumptions, restrictions and hypothesis.

C1. Añotador. Design and implementation of a temporal tagger for Spanish and English, for generic or legal texts. This result will be presented in Section 6.1.

C2. WhenTheFact. Design and implementation of an event extractor for European judgments that will be presented in Section 8.2.

C3. Corpora. Among the different resources developed, three freely available corpora with temporal information that are presented in Chapters 5 and 7 were annotated.

- **C3.1. TempCourt corpus**. Corpus of judgments in English from different courts manually annotated with temporal expressions. This corpus is the first of its kind available in the legal domain. It will be introduced in Section 5.1.
- **C3.2. HourGlass corpus**. Corpus of short texts in Spanish from different sources, countries and linguistic registries. This work was conceived to more efficiently test the capabilities of a temporal tagger in Spanish, targeting typical difficulties that are not present in the available corpora that are used in challenges. The corpus will be introduced in Section 5.2.
- **C3.3. EventsMatter corpus**. Corpus of judgments in English annotated with events. EventsMatter is presented in Chapter 7.

C4. ft3 Ontology. In order to represent judgments as a series of relevant events, an ontology supporting the storage of both temporal information and data related to its annotation has been developed. This resource will be presented in Section 9.1.

C5. Additional Tools. During the development of the thesis, additional tools to the ones presented above were created. Some of them can be used as standalone applications, while others were integrated into the ones previously presented or are somehow complementary. The main ones are outlined below.

- **C5.1. lawORdate**. Web service that finds legal citations that can be misleading to a temporal tagger in a text in Spanish and replace them with harmless expressions. Once the temporal tagging is done, the original citations are done. This service is introduced in Section 6.2, and an English adaption of it is used in the WhenTheFact Event Extractor (Section 8.2).

C5.2. ContractFrames. A software able to process a text in English detailing certain events about a contract that returns it in different formats, including RDF and PROLEG. It is briefly introduced in Section 8.1.

C5.3. Structure Extractor. Code able to extract the sections from European judgments. It is part of the event extractor WhenTheFact, detailed in Section 8.2.

C5.4. ft3 Converter. Online converter among different temporal annotation formats and the ft3 ontology, described in Section 9.2.

C5.5. Legal Event-Based Knowledge Graph. A knowledge graph populated with events from European legal decisions, presented in Section 9.3.

4.3 Assumptions

The work presented in this thesis is done under the following set of assumptions:

A1. It is assumed that the temporal expressions can be normalized to the TimeML standard (or the extensions proposed in this thesis).

A2. The concept of relevance is subjective and its definition highly depends on the ad-hoc application it is considered for. It was therefore decided to bound it by asking annotators to mark events they considered relevant enough to appear in a timeline that helps to understand the case.

4.4 Hypotheses

H1.a. Temporal Expressions in legal texts present particularities that are not covered by the current temporal taggers.

H1.b. Preprocessing the legal text in order to handle part of these particularities and directly targeting others in the temporal tagger would improve the results with regard to applying a generic temporal tagger.

H2.a. Legal texts contain different types of events, with different relevance and framed in different timelines.

H2.b. The creation of a data model that allows the representation of these events together with contextual information and their annotation details will allow facilitating the document representation in an event-based semantic way within the text.

H3. The integration of the event-extractor for the legal domain and the data model will allow the population of a knowledge graph, that can later facilitate alternative semantic representations based on events such as timelines, semantic searches or summarization generation.

4.5 Restrictions

The work presented in this thesis is subject to the following restrictions:

R1. The scope of the research will be restricted to a certain type of legal documents, namely judgments, due to the high amount of temporal expressions and the narrative structure.

R2. The source is also limited to European courts, since the availability is not as limited as in other jurisdictions.

R3. Although the information related to an event can be spread through different sentences, and also several events can be present in the same sentence, in this work just one event per sentence is considered.

R4. Similarly, coreference is not tackled in this work.

R5. This research covers texts in Spanish and English.

4.6 Research methodology

The research methodology followed in this thesis consists of three stages, one for each area to tackle. At the same time, each stage implies the analysis of the state of the art, the identification of the limitations, the proposal of solutions and their evaluation (testing them against other proposals in the state of the art). Additionally, a more general study of the state of the art was performed at the beginning of the PhD. The process followed and the tasks accomplished are detailed below:

The first phase of this thesis consisted of an **analysis of the state of the art** regarding the temporal annotation task and the representation of temporal information. This review covered the availability of tools, annotation standards, representation options and semantic resources (especially corpora), as well as evaluation options. From this in-depth examination of the state of the art, the definition of the **main gaps** in the domain was derived.

The second phase of the thesis covered the approaches to the different objectives and tasks to tackle. Each of these was addressed at a different stage.

Stage 1: Approach to the Temporal Tagging task. In this stage, the semantic resources (mainly corpora), annotations standards and representation options available were gathered and analyzed. From the lessons learned, additional tools (e.g., lawORdate) were created in order to mitigate the gaps detected during the first phase of the thesis, as well as corpora, one of the main gaps in the domain. Regarding the temporal tagging task itself, an evaluation of existing temporal taggers in the legal domain was performed, since there was no such evaluation on legal texts in literature. Concerning the Spanish language, an evaluation of the performance of the temporal taggers available targeting Spanish showed that they covered a limited portion of the ways to express temporal information. With these shortcomings in mind, a temporal tagger focusing on a broader coverage of the Spanish language and the legal domain was created and evaluated against different corpora and temporal taggers. Finally, these results were published in different journals.

Stage 2: Approach to the Event Extraction task in the legal domain. As in phase 1, an analysis of previous approaches in the domain and the different definitions of event was performed. Then, a corpus annotated with relevant events in legal judgments was created in collaboration with different legal experts. Different tools were built to properly process and visualize these annotations, and also a rule-based event extractor was created and evaluated. These results were presented in different publications.

Stage 3: Proposal of representation of temporal information. This final stage covered the representation of temporal information, derived from the lessons learned from the previous tasks, that led to the definition of the requirements. Afterwards, an analysis of previous representation options and how they covered the requirements was performed. Lastly, a data model that covers the needs of the temporal annotation task was created. This model, together with a temporal annotation format converter, was integrated into the previous software and was used for the creation of a legal event-based knowledge graph to demonstrate its usability.

Finally, the last part of the thesis comprises the publication of the final results and the writing of the dissertation.

4.7 Evaluation methodology

Regarding the hypotheses defined in this thesis, the following evaluations have been performed:

E1. (for H1). The goal of this evaluation is to test if a legal dedicated approach, taking into consideration legal particularities, improves the results by the state of the art temporal taggers. To this end, the results are compared using NLP metrics, which will be presented later in this section.

E2. (for H2). The goal of this evaluation is to determine the consistency and completeness of the data model built to represent events. To this end, it will be used to represent the events in texts and to convert between different existing formats.

E3. (for H3). The goal of this evaluation is to assess that the event extraction method is able to transform a legal judgment into a series of events that can be further used for other tasks. The whole pipeline necessary to do this will be built, creating an event-based knowledge graph using the data model, that will be queried to retrieve the extracted events.

The sections below present how the NLP tasks targeted in the thesis will be evaluated.

4.7.1 Temporal Expression Identification and Normalization

For the evaluation of the detection and normalization of temporal expressions during this PhD thesis, the typical *precision*, *recall* and *F-measure* metrics, which are commonly used in literature for this task (Strötgen and Gertz, 2012), were used. These metrics are explained below.

- **Precision** is the amount of correctly identified items divided by all identified items.

- **Recall** is the amount of correctly identified items divided by the items that should have been identified.

- **F-measure** is the weighted average between Precision and Recall.

Additionally, when dealing with text annotations in text, there is an extra nuance that can be considered.

- A measure is **strict** when it just considers as correct the annotations that match exactly the reference annotation. Partially matching annotations, that overlap but do not perfectly match the extent of the reference annotation, are therefore not counted as correct.

- A measure is **lenient** when it accepts partial annotations as correct.

The main reason to add this consideration to the evaluation is that, independently to the correct extent of the annotation, the normalization of a temporal expression can be correct. Also, in some cases, the correct extent of a temporal expression is not clearly derivable from the official guidelines. So if the temporal expression is correctly detected (even if the extent does not match exactly the human tagging in a corpus) and normalized, considering it incorrect would not be fair.

4.7.2 Event Identification

Regarding events, also the previously presented measures will be used when an annotated corpus is available. Nevertheless, it must be considered that since the task of annotating relevant events is subjective, the evaluation of the final software, working over texts extracted from sources that have not been used in corpora, the results cannot be evaluated merely regarding the figures. It must also be taken into account that false positives should not be evaluated as negatively as false negatives, since it is problematic not to find a relevant event but to consider an event relevant when it has not been tagged that way is in comparison a minor problem.

Summary

This chapter presented the main objectives and contributions of the thesis. It concludes the introduction of the dissertation, where I have also presented its context, reviewed the state of the art and analyzed the main challenges ahead.

The next chapter inaugurates Part II of the thesis, which presents the work done concerning temporal expressions.

Part II

TIME EXPRESSIONS

Chapter 5

Created Corpora

When the research on temporal information was conducted, several gaps in the area arose. One of the most critical obstacles in the way to develop software dealing with temporal information was the scarcity of corpora, both from the perspective of the legal domain and the Spanish language.

First of all, it was not possible to find any available annotated corpus of legal documents. Despite several previous works have analyzed the temporal dimension in legal documents, they were just theoretical or the corpora generated were not publicly available. In terms of document types, literature (Schilder, 2005) distinguished among different kinds of legal documents, namely transactional documents (this is, documents written by lawyers for specific transactions, such as contracts or agreements), constraints in statutes or regulations, and legal narratives in case law. The first two types of documents received dedicated attention, but narratives in case law were assimilated to narratives present in news and were therefore relegated to a non-legal specific approach. Later works tend to focus on one of these specific types of documents, such as done by Isemann et al. (2013) for European directives and by Guda et al. (2011) for dealing with temporal information in transactional documents, while narratives in case law remained without a specific approach.

Second, most corpora available are in English. Spanish corpora are scarce and do exclusively cover news or historical texts. When Añotador, the tool developed in this thesis for the identification and normalization of temporal expressions, was in an early stage, this lack of both corpora in Spanish and variety regarding the types of texts hindered notably its evaluation. A way to test the different types of temporal expressions that can be found in natural language was missed, not just to get a number against a specific text. In parallel, the testing of the temporal taggers available for the Spanish language showed that, despite obtaining good results in temporal tagging challenges in Spanish, most of them were no longer operative or did not cover properly frequent temporal expressions in Spanish.

Due to these circumstances, during this thesis, it was decided to build several corpora to cover the gaps in the field. The first of them was TempCourt, a dataset of thirty legal decisions in English from different courts manually marked up with two annotation sets of TimeML tags. This dataset is the first corpus publicly available in the legal domain and will be presented in Section 5.1. The second corpus created was called HourGlass, and is introduced in Section 5.2. HourGlass aims to cover two different functions when dealing with Spanish texts, and has therefore two parts. First, the SYNTHETIC part is a collection of 285 documents specifically designed to test some functionalities a temporal tagger should cover, such as detecting basic expressions like "It is five o'clock". In order to facilitate the use of the dataset (e.g., in the case that just the coverage of a specific type of expression needs to be tested), several tags such as "Hour", "Dates" or "False" (this is, sentences where some confusing expression should not be tagged) were added to each document. Second, the PEOPLE part is a collection of 67 documents (although just 63 of them were added to the final HourGlass corpus due to their ambiguity) proposed by people foreign to the temporal annotation task. They were asked to write sentences with what they considered to be temporal expressions, and these sentences were afterwards analyzed and annotated. Besides the tags, also the register of the sentence (e.g. "normal", "Latin American" or "colloquial") was added. This serves to test if a temporal tagger is actually able to target temporal expressions out of the target it is used to tackle - namely, news written in Castilian-Spanish, the most common kind of text evaluated in challenges - by using real-world expressions.

5.1 TempCourt corpus

The TempCourt corpus was conceived and developed during a three-month research stay at the Wirtschaftsuniversität Wien in 2017. The resource is therefore the result of a collaboration with researchers from there, namely Erwin Filtz, Sabrina Kirrane and Axel Polleres, as well as Víctor Rodríguez-Doncel from UPM. The corpus is freely available online[85] under a GNU General Public License 3.0, and the work leading to its creation, as well as the lessons learned, were captured in a journal publication (Navas-Loro et al., 2019a). Part of the information, data and images from that publication are reused in this section.

TempCourt was created to respond to a gap existing in the legal domain concerning temporal information annotations, and its objective is three-folded. First, that temporal information has been previously analyzed in literature, to the best of our knowledge there is no publicly available corpus of legal documents annotated with temporal expressions. Second, the lack of this corpus had hindered the evaluation of existing temporal taggers in the legal domain, as well as an empirical analysis of the particularities of legal texts with regard to the way to express temporal information. Finally, including both human annotations and the result of different state-of-the-art temporal taggers allows other researchers to compare their own approaches to existing tools. This is especially useful in NLP, since reproducibility of previous results is one of the main obstacles when trying to develop new solutions for a task. The TempCourt corpus is to this aim, not just a bunch of annotated documents, but also the first benchmark publicly available for temporal annotation in the legal domain. The corpus contains therefore different annotation sets, whose annotation methodology and processing will be detailed in the following subsections.

Section 5.1.1 introduces how the data was collected and describes the sources, justifying their selection. Section 5.1.2 explains how the manual annotations were performed. Section 5.1.3 describes how the output of different state-of-the-art temporal taggers was added to the corpus in order to facilitate the evaluation of new tools. Section 5.1.4 details the final format of the documents and how they have been made available to the research community. Finally, Section 5.1.5 shows some statistics on the corpus.

5.1.1 Data collection

Since there was no previous information about the annotation of temporal information in legal texts, our first step was to gather documents from three different legal sources in order to make a first attempt of annotation following the TimeML standard general guidelines. Although several different types of documents could have been chosen to create a first gold standard, we finally decided on legal judgments and preliminary

[85] `https://tempcourt.github.io/TempCourt/`

assessments of applications, since they tend to contain a vast number of temporal expressions.

Taking into account that most of the existing temporal taggers only support the English language, we selected court decisions in this language in order to enable a fair comparison among them. Furthermore, we did not want to constraint to a unique court, since this would restrict the variety of ways in which temporal information is represented. We, therefore, decided to collect documents from different countries and jurisdictions. Since not all the courts or organisms allow the collection and reuse of their documents, we finally opted for the following three courts:

1. European Court of Justice (ECJ): the highest court of the European Union, also known as the Court of Justice of the European Union (CJEU). The ECJ studies legal cases from the different member countries and guarantees that the European Union legislation is correctly interpreted and applied equally in every member country[86]. This is accomplished, among other procedures, via the *preliminary rulings*, that are decisions of the ECJ that respond to a specific request from a national court that is in doubt on how to apply to a particular case a part of the European Union legislation. The documents can be downloaded from the EUR-Lex database[87], and can be reused in conjunction with the Commission Decision of 12 December 2011 on the reuse of Commission Documents[88] for commercial and non-commercial purposes given the source is acknowledged[89]. Documents from this court have in fact been previously used in several legal artificial intelligence tasks in literature (Mencía and Fürnkranz, 2010; Quaresma and Gonçalves, 2010), although never in temporal tagging.

2. European Court of Human Rights (ECHR): also referred to as the Strasbourg Court, this court rules on individual or State applications alleging violations of the civil and political rights set out in the European Convention on Human Rights[90]. Differently to the European Court of Justice, the ECHR does not only cover European Union countries, but also other European countries, such as Russia.

[86] `https://europa.eu/european-union/about-eu/institutions-bodies/court-justice_en`
[87] `http://eur-lex.europa.eu/`
[88] `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32011D0833`
[89] `https://eur-lex.europa.eu/content/legal-notice/legal-notice.html#droits`
[90] `https://echr.coe.int/Pages/home.aspx?p=court`

When a case is submitted to the ECHR, some admissibility criteria are analyzed in order to decide if it should be judged by the court. The decisions of the ECHR can be found at the HUDOC database[91], and are allowed to be reproduced for private use or for the purposes of information and education in connection with the Court's activities when the source is indicated and the reproduction is free of charge[92]. This court is becoming a common source of documents in academia, both for teaching and for research (Medvedeva et al., 2020).

3. United States Supreme Court (USSC): also known as Supreme Court of the United States (SCOTUS) is the highest court in the United States and is charged with ensuring US citizens equal justice under law and interpreting of the Constitution[93]. The documents are available via the court's webpage[94], and since they are published by US governmental institutions, they are in the public domain[95].

As usual in legal documents, texts in our corpus contain the name of people, such as involved judges and parties, in a non-anonymized way. Names are considered personal data and are therefore object of the General Data Protection Regulation (GDPR)[96]. Nevertheless, Article 14 specifies that in the case of public data transparency must be provided with respect to the processing on request, and that consent for the processing of personal data from the data subject is not required for public data.

Each court also have its own structure for the documents. Section 3.1.3 details the different parts of the documents, that are summarized in Table 3.2.

Regarding the size of the corpus, we collected ten documents per court, for a total of thirty documents. The statistics of the corpus in terms of the number of tokens, document size and sentence length are detailed in Section 5.1.5.

5.1.2 Annotation Methodology

The methodology followed for building the TempCourt corpus is depicted in Figure 5.1. We first gathered documents from the three sources exposed in the previous section

[91] `https://hudoc.echr.coe.int`
[92] `https://echr.coe.int/Pages/home.aspx?p=disclaimer&c=`
[93] `https://www.supremecourt.gov/about/about.aspx`
[94] `https://www.supremecourt.gov/opinions/slipopinion/19`
[95] `https://www.copyright.gov/title17/92chap1.html#105`
[96] Regulation (EU) 2016/679.

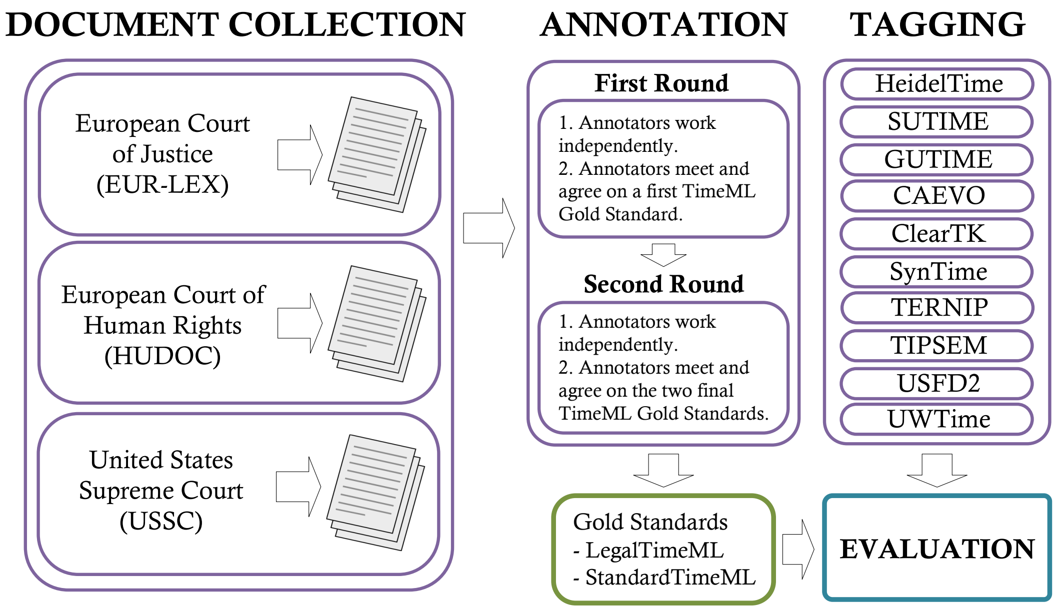


Figure 5.1: Outline of the building of the TempCourt corpus, including document collection, annotation and evaluation of state-of-the-art taggers.

and transformed them into an editable format. We then performed a first round of independent annotations by the two annotators involved in the creation of the corpus, following the TimeML standard official guidelines. This manual annotation was done using the GATE (Cunningham et al., 2013). Once completed, the two annotators met to create a gold standard with annotations agreed upon by both of them. When any difference was detected, or doubts arose, the TimeML guidelines or previously annotated official corpora were consulted specifically looking for similar cases. Some doubts were also referred to experts in the standard when the way to implement it was not clear, although no answer was ever provided. If after all these considerations the doubt persisted, also the TIDES TIMEX2 guidelines[97] were examined, as referred to in the TimeML annotation guidelines.

Besides this process targeted to fit into the TimeML standard, the annotators soon encountered legal particularities and domain-specific language issues that led to the definition of some additional decisions on the annotation. Since, nonetheless, we did not want to violate the standard, we decided to develop two different gold standards for the corpus, one of them following the TimeML guidelines and the other one taking

[97] www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-timex2-guidelines-v0.1.pdf

into account domain considerations. We detail below some of the legal particularities discussed after the first round of annotations:

1. Legal documents, and especially judgments, often contain references to previous court decisions or legislation in the legal grounding of a decision. The citation of such preceding cases or regulations depends on how decisions of such courts are usually referenced, and tend to include dates in them. Typically, at least a year is contained in the citation and annotated as a temporal reference, or they have to some extent a date-like format. We came to the decision that temporal information contained in identifiers used to refer to collections of court decisions (e.g. *2006*I) or included in the document identifier should not be annotated (e.g. EC:C:*2013*:180).

2. For the annotation of references to the present time, the TimeML standard (as well as the most extended temporal taggers) uses the *PRESENT_REF* token as a value to normalize references to present, as well as *PAST_REF* and *FUTURE_-REF* for the past and future ones, respectively. Other taggers, on the opposite, normalize these expressions to a date (usually the creation date). We decided that for the legal domain we should follow the latter approach, since all the documents in the corpus contain the information needed to reach this conclusion, and humans would also be able to derive it.

3. Expressions such as "*the date indicated*", appearing for instance in the excerpt "*the application lodged on* ***the date indicated*** *in (...)*" are not considered as temporal references but as co-references, being therefore not annotated in the gold standard, since a temporal tagger would not be expected to do so.

4. The word *now* is heavily used in legal documents with a non-temporal meaning. We, therefore, decided only to annotate it when not used as an adverb, being the meaning different to the temporal *currently* or *at the moment*. An example of this can be found in the case ECJ C-457/12, "*(...) so the provision is now worded as follows ...)*", where *now* can be omitted without changing the sentence in any way, acting like some legal tagline.

The discussion of these considerations between the two annotators, among others, resulted in the creation of two different gold standards: *StandardTimeML* and *LegalTimeML*:

1. **StandardTimeML** annotates all the TEs following the TimeML guidelines, including the use of *PRESENT_REF*, *PAST_REF* and *FUTURE_REF* tokens, as usually done by well-known temporal taggers.

2. **LegalTimeML** annotates just the TEs relevant to the narratives of the judgment, following the particularities in the legal domain previously discussed. It does not annotate, for instance, any dates part of a legal reference, and normalizes all the expressions to dates, when possible. As per the *StandardTimeML* annotation set, it follows the guidelines but does not annotate all the expressions, being, therefore, a subset considering domain particularities with slight differences in normalization.

After this core decision, the second round of annotations was conducted, again independently, and a last meeting conveyed to the two final Gold Standards described above. An analysis of these two final Gold Standards was done using the Inter-Annotator Agreement (IAA), which resulted to be high (0.95), as well as Cohen's kappa Cohen (1960) (0.94) and Scott's Pi Scott (1955) (0.94). This indicates that the normalization of the TE's that are included in both annotation sets have a high agreement. If we consider the differences between annotations from a quantitative perspective, we find there are an average of 13.1 common TEs per document, 0.3 partial coincidences and about 16.2 TEs that are not in the *LegalTimeML* but appear in the *StandardTimeML*. The recall among both annotation sets is of 0.44, while precision is 0.90. This confirms that a great amount of TEs are not relevant for the case timeline, that is what is considered in the *LegalTimeML* annotation set (44% with regard to the ones annotated following the full TimeML standard-compliant annotation set *StandardTimeML*), but that the way to tag them by the annotators is highly similar.

5.1.3 Corpus as a Benchmark

Once the corpus was collected and the two manual annotation sets were produced, we decided to test ten state-of-the-art temporal taggers on it. The objective of this endeavour was two-folded. First, storing the annotation of several temporal taggers along with the corpus would help eventual researchers to compare their approaches to previous ones, regardless if they are still operative or not, facilitating also the task, since it avoids the need for preinstallation of any required software. Secondly, this tagging allowed to perform the first empirical analysis of how well existing temporal taggers operate on legal documents from several jurisdictions. Section 5.1.3.1 explains how these state-of-the-art temporal tagger's annotations were added to the corpus , while Section 5.1.3.2 presents an analysis of the main difficulties found by them.

5.1.3.1 State-of-the-art Annotations

Different state-of-the-art temporal taggers, which represent the different approaches existing in literature and have been widely used before, were applied in our corpus. All of them were described in detail in Section 2.3.1, and are namely the following: HeidelTime (Strötgen and Gertz, 2012), SUTime (Chang and Manning, 2012) GUTime (which is part of the TARSQI toolkit) (Verhagen et al., 2005), CAEVO (Chambers et al., 2014), ClearTK-TimeML (Bethard, 2013), SynTime (Zhong et al., 2017), TERNIP (Northwood, 2010), TIPSem (Llorens et al., 2010), USFD2 (Derczynski and Gaizauskas, 2010) and UWTime (Lee et al., 2014). These ten temporal taggers were executed over our legal corpus with diverse results over the documents from the three different sources. Some of them included different parametrization options or ways to be called, so we detail below the considerations taken into account, in case the annotations need to be replicated:

1. **HeidelTime** was called using its narrative text function; this tool also offers specific approaches to other kinds of texts, such as news, but we considered that this one was the more suitable for the task.

2. **GUTime** was used as a part of the TARSQI toolkit and was called alone with the preprocessor in the pipeline.

3. **USFD2** was slightly modified in order to annotate any input and to generate TIMEX3 tags as output, since the code available online was just able to annotate a specific corpus. Nevertheless, its functionality and the rules it applies were not modified.

4. All the taggers were called via a Java application that used the library or ran the command line needed in Windows, except of **TIPSem**, which required a special environment and preinstallation and was called via command line in a different Linux virtual machine.

5. All other taggers were used with default parametrization.

Regarding the output, despite all the taggers following the TimeML standard, not all of them implemented it in the same way. Most tags were added inline (this is, in the text annotated), but the TARSQI toolkit produced them offline, at the end of the document and referring to their position in the text using offsets. A specific converter to transform the offline generated output into inline one was coded to resolve this difference since all outputs had to be in the same format to be properly integrated into the same file to compare and store them. Once the outputs of all the taggers were in the same format, a second routine allowed the creation of a GATE document that contained the different annotations of the ten temporal taggers, as well as the two manual annotation sets (*StandardTimeML* and *LegalTimeML*), excluding any other additional non-temporal expression annotation produced by the temporal taggers. The results of this automatic annotation process are discussed below.

5.1.3.2 Main difficulties found

The thorough analysis of the corpus documents and the manual inspection of the most frequent errors of the taggers led to the synthesis of a collection of test cases that present the phrases prone to cause errors. Information on the specific mistakes made by each tagger can be found in the publication about the corpus (Navas-Loro et al., 2019a). Below we list some of the most common errors in which the taggers fall, whether because they happen frequently in the text or because several taggers incur in them.

- Dates expressed in the format "DD/MM/YYYY", frequently provoke problems in identification, and in some cases also in normalization (specially when dealing with international dates, that can have formats such MM/DD/YYYY or even YY/MM/DD).

- Identification of a currency as a year ("*EUR **2000***").

- Tagging of expressions such as "*62-year-old*".
- Splitting of whole `SET` expressions as "*Once a week*" into "*Once*" and "*a week*", converting one `SET` into a *PAST_REF* `DATE` on one hand (*Once*), and a `DURATION` on the other (*a week*).
- Tagging always general ambiguous expressions such as "*fall*" or "*may*", that depending on the context might not have a temporal meaning. On the other hand, in legal texts we also find specific ambiguous expressions such as the case of "*dec.*", a non-temporal expression that appears when citing *decisions on admissibility*[98] and means "decision", despite it is commonly misinterpreted as "December" by the temporal taggers.
- Separation of `DURATION`s such as "*One year and one day*" into two different `DURATION`s. This is probably due to the lack of these kinds of expressions in non-legal texts.
- Not recognizing series of `DATE`s such as "*15 and 16 December*", but detecting the last `DATE` of such a series only (because this is usually the most complete one). Differently from the previously reported error, dealing with durations, this one was particularly noteworthy, since the series of dates are not at all exclusive from legal texts.
- In some documents (as also happens in other kinds of legal texts, such as transactional like contracts), some information is put into brackets, such as in "*before the expiry of a period of* ***[48] hours***"; usually generic temporal taggers are not able to detect them (for instance tagging in this concrete example just "*hours*").
- Tagging year-like expressions such as "*No* ***1612****/68*" or "*§****1408***"; most taggers in fact tag every four-digit number as a year.
- Most taggers do not take modifiers (`mod`) such as *EQUAL_OR_LESS*, *LATE*, *END*, or *EARLY* into account. Probably this responds to the low ratio of appearance of them in other domains, even though they are extremely important in legal documents.

[98] `http://www.echr.coe.int/Documents/Note_citation_ENG.pdf`

- The case of the `quant` and `freq` attributes is similar for `SET`s, which is usually ignored by general temporal taggers despite their importance in the legal domain.

5.1.4 Format

The final documents have been generated in several formats[99]. First, as GATE XML documents, that facilitate the storage of different annotation sets and also the visual and numerical comparison of the different sets. Second, a set of TimeML documents (TML) is provided for each of the manual gold standards. These documents contain the same annotations as in the correspondent annotation set in the corresponding GATE document, but makes the comparison with the output of other temporal taggers easier, as it is in the *official* TimeML format. Also a set of TML documents without any tag is provided to facilitate testing. These TML documents have been validated using the TimeML validator from TempEval-3[100], so it is guaranteed that they fulfill the guidelines of the TimeML standard. Finally, all original documents are stored as TXT files; the size of these documents in terms of kilobytes and their length in tokens are shown in Table 5.1.

Corpus	**# Doc.**	**# Tokens**	**Doc. Size (Avg. KB)**	**Doc. Size (Avg. Tokens)**	**Sentence length (Avg. Tokens)**
ECHR	10	7,252	4	725	13
ECJ	10	53,044	32	5,304	32
USSC	10	50,874	25	5,087	18
Total	30	111,170	20	3,705	21

Table 5.1: TempCourt corpus statistics.

5.1.5 Statistics on the Corpus

Table 5.1 presents the statistics of the corpus, where the differences between the three source courts become evident. The documents in the ECJ and USSC subcorpora are similar in terms of document size and length, while the documents in the ECHR subcorpus are only one fifth in terms of size in comparison with the other two subcorpora.

[99] The final corpus can be downloaded at `https://tempcourt.github.io/TempCourt/`

[100] `https://web.archive.org/web/20200811064918/https://www.cs.york.ac.uk/semeval-2013/task1/`

This makes ECHR documents much simpler to be understood, both from the automatic processing point of view and from the layman perspective. It must be also taken into account that, as stated previously, legal texts often make use of very long and complicated sentences to explain legal details, thus we also included the average sentence length in tokens for each corpus. We can also appreciate in the table that the sentences of the ECHR are around one third of the length compared to the USSC court decisions, and also tend to be shorter than the ones in the ECJ corpus. These numbers contrast with those relating to corpora from other domains and sources, such as Wikipedia articles (25.1 words per sentence (Kajiwara and Komachi, 2016)), the CONLL 2007 corpus of documents from the Wall Street Journal (24 and 23.4 tokens per sentence in training and test data, respectively (Nivre et al., 2007)) and the basic corpus of everyday documents (Pellow and Eskenazi, 2014), including all kind of common texts, such as banking or shopping documents (with an average of 17.2 words per sentence). Regarding the number of documents in each corpus, Table 5.2 provides an overview (extracted from previous literature (UzZaman et al., 2013)) of the size of referential corpora manually annotated with TimeML. Despite one of the comments we received from the reviewers of the publication derived from this work was that the corpus might seem too small to be considered representative enough, the figures in Table 5.2 show otherwise. These numbers illustrate differences in amounts of documents and tokens of previous temporally annotated corpora depending on the source. Therefore, all these figures provide evidence that despite not having a lot of documents, our corpus is substantially bigger in terms of tokens than most of the previous corpora.

Corpus	**# Doc.**	**# Tokens**	**Doc. Size (Avg. Tokens)**
TimeBank[101]	183	78,444 (61,000[102])	428.7
AQUAINT[103]	73	34,154	467.9
TempEval-3 Platinium Eval. (UzZaman et al., 2013)	20	~6,000[104]	~300
WikiWars (Strötgen and Gertz, 2016)	22	119,468	5,430.4
Time4SMS (Strötgen and Gertz, 2016)	1,000	20,176	20.2
Time4SCI (Strötgen and Gertz, 2016)	50	19,194	383.9
TempCourt (Navas-Loro et al., 2019a)	30	111,170	3.705

Table 5.2: Statistics of corpora annotated with TimeML in literature, including the created resource TempCourt.

Regarding the annotations, the average number of annotations per corpus in both Gold Standards (LTML and STML) and the various taggers are shown in Table 5.3, which illustrates the occurrences of different TIMEX3 annotation types (`DATE`, `DURATION`, `TIME`, `SET`) for each analyzed corpus. It is clearly shown that the most used annotation type in court decisions is DATE. This result is not surprising as the date is considered to be sufficient in most cases as the actual time of the day is not relevant. Furthermore, deadlines in the legal domain usually indicate the end of the day and it is not important if an action is taken in the morning or in the afternoon. Also, the fact that the pattern of appearances of each of the TIMEX3 types does not fit any of those of the domains analyzed by Strötgen and Gertz (2012) (news, narratives, colloquial and scientific) must be noted.

	ECHR				**ECJ**				**USSC**			
Tagger	**D**	**Dur**	**S**	**T**	**D**	**Dur**	**S**	**T**	**D**	**Dur**	**S**	**T**
StandardTimeML	11.6	1.3	1	0	31.5	4.3	2	2.7	35.7	5.6	3.5	4
LegalTimeML	10.1	1.3	1	0	16.8	4.3	1.5	3	9.1	5.4	1.5	0
HeidelTime	11.4	1.7	1	0	68.1	5.3	1	1	41.6	5.6	1.5	2
SUTime	11.3	2	0	0	39.1	3.9	1.3	1.3	46.9	7.9	1.5	2.7
GUTime	11.7	0	0	0	31.4	1	0	0	37.3	2	0	0
CAEVO	11.1	1.8	0	0	36.7	5.8	1	1.5	39.9	9.4	1.5	3
ClearTK	10.2	1	0	0	38.6	3.4	0	0	36.1	5.1	1	2
Syntime	11.5	0	0	0	39.1	0	0	0	47.8	0	0	0
TERNIP	11.7	1.7	0	0	30.3	3.6	0	0	33.3	5.6	1	0
TIPSem	13	1	0	0	38.4	2.6	0	0	-	-	-	-
USFD2	13.9	2	0	0	66.6	3.3	0	0	28.4	3.8	0	0
UWTime	11	2.5	0	0	-	-	-	-	-	-	-	-

Table 5.3: Average number of annotation types per document for each corpus (***D**ate*, ***Dur**ation*, ***S**et*, ***T**ime*).

[101] `https://web.archive.org/web/20160804062727/http://timeml.org/timebank/documentation-1.2.html`

[102] The website mentions 61k non-punct tokens, 78k was extracted from Strötgen and Gertz (2016).

[103] `https://web.archive.org/web/20160804062727/http://timeml.org/timebank/aquaint-timeml/aquaint_timeml_1.0.tar.gz`

[104] Just approximate figures were provided (UzZaman et al., 2013).

5.2 HourGlass corpus

The HourGlass corpus is a benchmark for temporal taggers in Spanish. The corpus is freely available online, both on a website with further information[105] or via Zenodo[106], under a GNU General Public License 3.0. This work was presented in the conference Language Knowledge and Engineering, held in October 2019, and later published as a journal publication (Navas-Loro and Rodríguez-Doncel, 2020). Part of the information, data and images published there are reused in this subsection.

While the first version of the software Añotador was being developed (see Section 6.1), the scarcity of corpora in Spanish became evident, despite its importance in the world[107]. When analyzing the available corpora, we also found several problems that needed to be covered. To this aim, we created the HourGlass corpus. The reasons are summarized below.

Besides the scarcity of corpora in Spanish, we also noticed that available corpora tend to comprise just some specific kinds of texts, namely news and historical texts for Spanish, and does therefore not cover all the possible variance in which temporal information can appear in natural language.

At the same time, having the temporal expressions not categorized in any practical way regarding what they represent does not facilitate the testing of systems aiming for their detection and normalization. From a developer perspective, when we add new rules or retrain a temporal tagger to increase its coverage, we risk stopping correctly covering expressions that we did correctly before the changes. We consider that a corpus trying to cover this gap should therefore take into account these practical considerations.

Finally, regarding corpora in the Spanish language for temporal tagging, just texts in Castilian Spanish have been found in literature, leaving aside other Spanish-speaking countries, although the Spanish spoken in Spain is not even the most frequently found around the world[108]. Other countries express basic temporal expressions in a different

[105]http://annotador.oeg.fi.upm.es/hourglass.html

[106]https://doi.org/10.5281/zenodo.3415633

[107]According to Instituto Cervantes, a worldwide non-profit organization created by the Spanish government in 1991 responsible for promoting the study and the teaching of Spanish language and culture, Spanish is spoken by more than 580 million people in the world, around 483 million of them being native. Spanish is, therefore, the second language in terms of international communication, being also the most spoken one whose amount of speakers is increasing (Cervantes, 2019).

[108]According to Instituto Cervantes (Cervantes, 2019), 42,915,985 speakers from Spain are native speakers, while Mexico and Colombia have 121,899,691 and 49,436,235, respectively. Almost

way than Spain's way to do it, and this should be considered in a corpus. An analogous problem happens regarding the register. Existing corpora just cover regular written texts, excluding informal or cultured language, that include special ways to deal with temporal expressions.

For all these reasons, a corpus specifically designed to facilitate systematically testing temporal taggers in Spanish was built. The HourGlass corpus includes expressions from different registers and countries, and each text in it is clearly categorized in order to allow the partial testing of temporal taggers' different capabilities.

Section 5.2.1 details how the texts in the corpus were gathered. Section 5.2.2 presents the different tags used to classify the texts in the dataset, as well as the annotation process. Section 5.2.3 explain how the tagging done by the taggers were added to the dataset in order to facilitate its use for temporal tagger testing, while Section 5.2.4 details the format of the corpus. Finally, Section 5.2.5 shows statistics on the corpus, such as the amount of each type of annotation and POS information.

5.2.1 Data Collection

As introduced previously, the HourGlass corpus is divided into two parts clearly delimited, called the *synthetic* part and the *people* part. We will elaborate in this section on how these different parts were conceived and collected.

5.2.1.1 Synthetic part

The *synthetic* part is a collection of 285 short texts specifically designed to test some functionalities a temporal tagger (like Añotador) should cover, such as detecting basic expressions like "It is five o'clock". They are therefore written explicitly to detect fails in temporal taggers when dealing with specific temporal phenomena. The texts were written by a Spanish speaker who is familiar with the TimeML standard and the task of temporal tagging, being therefore aware of main gaps in the task, and also to Spanish language particularities when dealing with temporal expressions.

Additionally, tags such as "Hour", "Dates" or "False" (this is, sentences where some confusing expression should not be tagged) were added to each document in order to

400,000,000 native speakers come from Latin America, whose countries have more similar dialects, in comparison to Castilian Spanish.

facilitate their use (e.g., in the case the coverage of just some type of expression needs to be tested). More information about these tags will be provided in Section 5.2.2.

5.2.1.2 People part

The *people* part is a collection of 67 documents (although just 63 were added to the final HourGlass corpus due to their ambiguity) proposed by people foreign to the temporal annotation task. Twelve professionals from several disciplines, ages and Spanish-speaking countries provided time expressions for this part of the corpus.

They were asked to write sentences with what they considered to be temporal expressions, and add them to a spreadsheet. They also had the option to report them in any means of communication during a period of two weeks, which provoked that occasionally, during a random conversation, a contributor said some sentence and considered that it had a temporal meaning and should therefore be added to the corpus. Therefore, some of these sentences were thought specifically for the corpus, but others are sentences they used during real conversations and chats and that they asked to be included. This organic way to enrich the corpus helped to have a very variate register within the texts.

All the sentences were afterwards analyzed and annotated. Besides the tags, also the register of the sentence (e.g. "normal", "Latin American" or "colloquial") was added. Our volunteers were given a basic call for expressions including some examples of the four types of time expressions in the TimeML standard. During this process, we found out that while volunteers find more or less intuitive other tasks in NLP (for instance, what is a *named entity* is usually more or less clear to people out of the domain), they find it difficult to distinguish what is a temporal expression and what is not. Most of the expressions we got from our contributors are not envisaged in the standard, and should probably be marked as SIGNALS (the tag in TimeML used to mark expressions with some temporal information but that are not time expressions per se) or temporal relations. For this reason, some texts in this part of the corpus have no annotations. Other texts were too ambiguous to be annotated following the standard, and were therefore not added to the corpus, although we made them available to foster discussion.

5.2.2 Classification and Annotation

Once the documents from both parts of the corpus were collected, they were annotated and classified for being later used for testing purposes. In the final corpus, we included 348 documents, 285 synthetic texts and 63 from our contributors – more information can be found in Table 5.6. Four additional texts from our contributors whose annotation was ambiguous were not included in this final corpus (although we made them available along with the corpus).

Then, we first checked the sentences, especially the ones by contributors, in order to confirm that they could be annotated following the standard – the ambiguous expressions were marked and left aside. Also, we added comments for some of them and proceeded to their classification.

5.2.2.1 Classification

Since one of the reasons to build the HourGlass corpus was to facilitate the test of tools covering specific expressions, the first task tackled was the formalization of the tags of each part of the corpus.

In the case of the *synthetic* part, we normalized the different comments of each text (such as "to check if this way of expressing dates is covered" or "should not be tagged") into a set of more than twenty tags in order to facilitate testing. Some of these tags are for instance *Dates*, *Fractions* or *False* (referring to *false positives*, such as the expression "50/2/1991", that should not be tagged despite of looking like a date). These tags are especially useful in case we update a temporal tagger and we want to check if our coverage of certain time expressions changed, and also if we are interested just in some type of expressions.

For the *people* part, we normalized the tags differently, including for instance the tag *standard* if the sentence would be covered by the TimeML standard, *yes* if the text should be tagged but it is not clear how following the standard TimeML, *no* if it is an expression involving "temporal words" but where their meaning changes (or if it should not be tagged despite the contributors of the corpus thought that the text included some time expression), and *special* if it has some special meaning –an analysis of some examples of time expressions tagged as *special* can be found in Section 6.1.4. Besides the tags, we also added the register of each sentence. Among this register we find *colloquial*, *chat* (these sentences were extracted from chats, so grammar is less strict), *latin expressions* (very common in legal texts), *Latin American expressions*, *phrases* and one *literary* sentence.

A complete list of the tags typifying the text in the corpus, along with statistics, can be found in Table 5.4. Regarding the register, Table 5.5 shows the same information for them; please take into account that, while the *people* part of the corpus present all kinds of registers, in the *synthetic* part of the corpus the register of all the documents are categorized as *normal*.

Tag	Description	Synth	Pple	Tot
Ambiguous	Not clear how the expression should be normalized.	1	4	5
Century	Mentions to centuries or similar ("*Este siglo XX*")	10	-	10
Consecutive dates / Seq.	Long dates or series of them that might require to combine information to normalize ("*2, 3 y 20 de febrero de 2018*").	14	-	14
Correference	Expressions that refer to previous expressions in the text.	1	-	1
Date	Basic dates in several formats ("*19-02-1991*", "*3 de abril*")	26	-	26
Day	Reference to a day ("*el tercer día*")	1	-	1
Demostrative	Expressions with some kind of modifier that affects to normalization ("*Unos pocos días*")	3	-	3
Duration	TEs of the type duration ("*Duró seiscientos días*")	38	-	38
Expression	Common expression that has no temporal meaning, despite of using temporal words ("*Fue un giro de última hora*")	1	1	2
False	Imposible dates or misleading expressions ("*19/42/1991*")	14	3	17
Fractions	TEs including fractions ("*Tres horas y cuarto*")	13	-	13
Granularity	Expression with an specific granularity ("*aquel mes*")	2	-	2
Hours	References to time or hours ("*Las 12h*", "*De 12 a 3*")	36	-	36
Indef	TEs whose normalization is not defined or should be normalized using external information ("*Varias horas*")	11	-	11
Interval	Texts including intervals ("*Entre 1939 y 1945*")	5	-	5
Misc	Miscelanea ("*En el año mil*", "*En ese momento*")	8	-	8
Month(s)	Months in letters that might be confusing (in Spanish, *Abril* and *Julio* can be a month or the name of a person)	3	-	3
No	Expressions suggested by contributors that, according to the TimeML standard, should not be annotated ("*Ya estoy*")	-	19	19
Problems with past	Texts including the word "pasado", that is polysemic in Spanish.	6	-	6
Problems with sec.	Texts including the word "segundo", that is polysemic in Spanish.	1	-	1
Quarter	References to quarters, semesters, or similar.	1	-	1
Reference	Mentions of the past, present or future.	5	-	5
Relative	Expressions whose normalization should be anchored to a reference date ("*Anoche*", "*El próximo mayo*")	43	-	43

Set	Temporal expressions of the type set.	20	-	20
Special	Special ways of expressing time ("*Tiene 25 primaveras*").	-	3	3
Standard	TEs from the people part covered by the standard.	-	23	23
Time	References to parts of the day (Noche, tarde)	5	-	5
Trimester	Reference to semesters in a year ("*El tercer semestre*").	2	-	2
Weekend	Mentions to a weekend, as a date, duration or set ("*Dos fines de semana*", "*Cada dos fines de semana*").	4	-	4
Year	Mention to years ("*En el año dos mil*", "*A finales de 2019*").	11	-	11
Yes	Expression with special registers that should be annotated.	-	14	14

Table 5.4: Statistics of the tags characterizing the texts in the HourGlass corpus. The first column indicates the tag, described in the second column (sometimes including examples). The last columns are the amount of texts classified with those tags in the *synthetic* part of the corpus (third column), the *people* part of the corpus (fourth column), and the total amount (fifth column).

Register	Description	People
Chat	Text were extracted from chats (e.g. Slack), including shortcuts and grammatically flexible ("*Calculo 20:30*", "*lo vuestro dura 1h, no?*").	10
Coloquial	Ways to express time that are often used in daily talks ("*en cero coma*")	24
Latin	Expressions ot temporal information using latin terms ("*ipso facto*")	4
LatinAmerica	Texts including TEs used in Latin American Spanish countries ("*cinco para las 11*", that in Castilian Spanish would be "*once menos cinco*").	2
Literario	References extracted from books.	1
Normal	Texts that tend to reflect real situations, but are not as organic as other documents in the *people* part of the corpus.	24
Phrase	Phrases that should not be tagged because of the lack of temporal meaning (for instance, "*Hasta el 40 de mayo no te quites el sayo*" is a Spanish phrase where the "40th day of May" is mentioned).	2

Table 5.5: Statistics of the registers of the texts in the HourGlass corpus. The first column indicates the register, described in the second column (sometimes including examples). The last column provides the amount of texts classified with those tags in the *people* part of the corpus (since all texts in the *synthetic* part of the corpus are considered as *normal*).

5.2.2.2 Annotation

Once the tags were added, we started the annotation process. To facilitate this task, we first used a temporal tagger on the texts and we then manually corrected its annotations and added any missing ones. In order to avoid bias, we did not use our tagger for this task, but HeidelTime. Then, we performed the first round of annotations based on the

TimeML guidelines available for Spanish (Saurí et al., 2010).

Afterwards, the second round of annotations was done, reviewing the previous ones and correcting them when needed. The same anchor date ("2019-12-20") was used for the annotation of all the documents.

5.2.3 Corpus as a Benchmark

In addition to the manual annotations in the corpus, we also made available the annotation results of several temporal taggers. Section 5.2.3.1 introduces the tools used.

5.2.3.1 State-of-the-art Annotations

Similarly than done for the TempCourt corpus, we tested different temporal taggers on the HourGlass corpus in order to establish a benchmark for future temporal taggers. Nevertheless, not many tools cover the Spanish language.

The temporal taggers used were HeidelTime and SUTime. HeidelTime was called using the following parameters: "News" as the type of text, "Spanish" as language and "TreeTagger" as POS tagger. SUTime was invoked directly, not via the NER Annotator, as in the example code available in its documentation[109], but using the Spanish properties. Although we also tried to evaluate the temporal tagger TIPSem running it on different machines and configurations, we were never able to use it to process Spanish texts (despite we succeeded for English) due to the unavailability of some auxiliary software required by TIPSem. Additionally, the results of the first version of the software Añotador, introduced in Section 6.1, are also available with the corpus.

5.2.4 Format

The corpus is published (under a GNU GPL-3.0 license) as plain texts without annotations, TimeML files without annotations and TimeML files with annotations[110], along with excel files including the metadata of each document (namely, id, content, annotated text, if it belongs to the *synthetic* part or to the *people* part, tag, register and comments on the annotations)[111].

[109] `https://nlp.stanford.edu/software/sutime.html`

[110] Published in Zenodo http://doi.org/10.5281/zenodo.3415633

[111] To facilitate testing just one part of the corpus, files are named with their id, five numbers. If the first number is a 0, the file belongs to the *synthetic* part, while a 9 means it is from the *people* part.

The corpus and additional information about it can be found on its website[112], along with the result of the different taggers tested on it.

	synthetic		people		all	
	total	avg	total	avg	total	avg
documents	285		63		348	
sentences	292	1.03	67	1.06	359	1.03
TIMEX3	341	1.20	58	0.92	399	1.15
DATE	165	0.58	25	0.40	190	0.55
DURATION	102	0.36	21	0.34	123	0.35
SET	26	0.10	4	0.06	30	0.09
TIME	48	0.17	8	0.13	56	0.16
tokens	1927	6.76	688	10.92	2615	7.51
adjectives	69	0.24	20	0.32	89	0.26
adverbs	69	0.24	35	0.56	104	0.30
nouns	332	1.17	125	1.99	457	1.31
NPs	25	0.09	16	0.25	41	0.12
verbs	155	0.54	112	1.78	267	0.77

Table 5.6: Statistics on the occurrences of different types of temporal expressions and words in the HourGlass corpus, overall and for each part. They are given as total (e.g. the amount of tokens in the whole *synthetic* corpus) and on average (average of tokens per document).

5.2.5 Statistics on the Corpus

The statistics of the corpus are detailed in Table 5.6. There we can see how most of the temporal expressions covered in the text are of the type DATE, followed by DURATIONs. Also, information on the size of the corpus and the length of the texts included, along with the amount and the types of tokens, can be found.

[112] `https://annotador.oeg.fi.upm.es/hourglass.html`

Summary

This chapter presented the textual resources developed in the thesis with regard to temporal expressions. On the one hand, as a result of different collaborations a first corpus annotating and analyzing temporal expressions in legal narrative texts, covering a gap in the domain, was produced. On the other hand, a dataset in Spanish that covers different registers and Spanish dialects, and that is organized in a way that facilitates systematic testing of temporal tagging, in contrast to resources previously available, was gathered.

The next chapter will introduce the tool developed to handle some of the gaps detected in temporal tagging. Additionally, Chapter 7 will present the resources created dealing with events in the legal domain.

Chapter 6

Temporal Tagging

Temporal information is crucial in knowledge extraction. Being able to locate events in a timeline is necessary to understand the narrative behind every text. The extraction and processing of temporal expressions in textual documents have been extensively studied in texts such as news or clinical documents; however, for the legal domain, these tasks remain an open challenge. Also, most efforts have been done for the English language, and are usually constrained to the different challenges in the temporal tagging domain (e.g. TempEval) and the scarce corpora available.

In this section, the work tackling these gaps in temporal tagging is presented. First, Section 6.1 introduces *Añotador*, a rule-based temporal tagger able to process texts both in Spanish and English. The tool has two modes of use, one for generic texts and another one specifically designed to meet certain needs in the legal domain. Besides this legal dedicated implementation, Añotador also aims to cover some of the general shortcomings detected for the Spanish language, such as difficult-to-normalize commonly used words.

Section 6.2 presents *lawORdate*, a tool that tackles one of the particularities of the legal domain introduced in Chapter 3, namely the misleading legal references that have date-like format. lawORdate detects these expressions and replace them with innocuous expressions in order to make the text more temporal tagger friendly. Once the text has been annotated by a temporal tagger, lawORdate can be called in order to restore the references.

Finally, Section 6.3 presents an analysis of main challenges in the time expression extraction task.

6.1 Añotador

Añotador is a temporal tagger able to process both Spanish and English texts. Additionally, it works for two different types of texts: general texts (using the *standard* option of the tool) or legal texts (*legal* option of the tool). Although the general structure of Añotador is equal for both options, the implementation, the rules and the algorithm are slightly different, as well as the information tagged and normalized by each of the implementations.

Añotador works in two different phases. First, a rule-based system that operates over a Stanford CoreNLP pipeline[113] identifies temporal expressions. This pipeline includes a tokenizer, a sentence splitter, a lemmatizer, a POS tagger, a Named Entity Recognition tagger and the TokensRegex (Chang and Manning, 2014), a framework for defining cascaded patterns over token sequences where customized rules for time expression recognition are used. For these different needs, the default models for English from CoreNLP are used, since they are state-of-the-art tools in their respective tasks.

On the contrary, as of fall of 2021, some of the CoreNLP tools for Spanish were not efficient enough for the task; namely, the lemmatizer returned exactly the same token it analyzed, returning therefore not the real lemma. For this reason, it was decided to look for a better alternative, opting finally for IxaPipes (Agerri et al., 2014), and included the lemmatizer and the POS tagger models, trained with the Perceptron on the Ancora 2.0 corpus and more capable to process texts in Spanish. To this aim, part of an available code from the Fondazione Bruno Kessler[114], whose license (GNU General Public License (GPL) v3) allows modification under attribution, was adapted.

Thus, the implemented rules are applied to the output of the previous annotators in the last stage of the CoreNLP pipeline. In the second phase, the normalization algorithm determines the value of each of the expressions discovered by the rules and outputs them in the requested format (TIMEX3 or JSON).

Figure 9.10 depicts the pipeline of Añotador, which requires as an input just a text and optionally an *anchor date* (i.e., the reference date for normalization). This is, if the anchor date is "2019-05-20" and the expressions "el mes de marzo" ("the month of

[113] `https://stanfordnlp.github.io/CoreNLP/pipelines.html`
[114] `https://github.com/dhfbk/spanish`

March" or "el diciembre pasado" ("last December") are found, the normalization will be "2019-03" and "2018-12", respectively (see Table 1.1 for more examples).

The following sections will describe in more detail the operation of Añotador. Section 6.1.1 presents the rules developed for TE identification, while Section 6.1.2 introduce the normalization algorithm.

6.1.1 Rules

Añotador relies on a set of more than 200 iterative rules. These rules are token-based; this is, they take into account tokens instead of strings –this allows to consider information such as POS tagging or lemmatization. These rules are applied via the Stanford CoreNLP TokensRegex, where there are different types of rules. In the system the following ones are used:

- Token rules: they work on the token level and are applied at different stages, relying on information tagged by previous annotators in the pipeline (such as lemmas or POS) or previous rules. For instance, for the expression "dos días", meaning "two days", the rules would, first of all, annotate that "dos" is a number and that "días" is a type of temporal *granularity*. We call *granularity* to expressions such as "day", "month" or "century", that denote a specific way to measure periods of time. In a subsequent stage, Añotador is able to detect all the temporal expressions compound by the sequence "number + some granularity").

- Composite rules: differently than tokens rules, that are applied just once each, these rules work iteratively on tokens rules and on previous composite rules until there are no more matches.

These rules may produce several different actions, namely *annotations* (this is, internal tags that will be used by subsequent rules) and *results* (the final expressions with a specific set of values that will be returned to the normalization algorithm for further normalization). In the case of Añotador, the values returned are:

1. The type of expression, among DATE, TIME, SET and DURATION, the types envisaged by the TimeML standard. DATE refers to calendar expressions such as "October", "December 4, 2019" or "the first quarter of the year". TIME covers

clock expressions such as “one o’clock” or “tomorrow at 11pm”. SETs are expressions that repeat over time, such as “monthly” or “two days a week”. Finally, DURATION is the type denoting periods of time, such as “one week and a half” or “two days and three hours”.

2. The normalized *value*, that might require further normalization or not.

3. The *freq*, meaning frequency, in case of temporal expressions of the type SET (otherwise it will be empty). This value describes the frequency with which a SET expression is repeated (e.g., one month for the expression “monthly”).

4. The *mod*, meaning modifier, if there is any. Modifiers are optional, and the ones covered are those included in TimeML (namely BEFORE, AFTER, ON_OR_-BEFORE, ON_OR_AFTER, LESS_THAN, MORE_THAN, EQUAL_OR_-LESS, EQUAL_OR_MORE, START, MID, END and APPROX).

5. The last rule applied, so the reasoning that produced the result can be traced for debugging purposes.

In the following subsections how the rules are applied and how these results are generated via the intermediate annotation tags will be detailed.

6.1.1.1 Basic tokens

In the first stage, Añotador detects basic token-based relevant expressions, such as:

- Numerals: either expressed with numbers or words. This is a non-trivial task, as standard NER systems and POS taggers do not recognize numerals when represented with words, as in the example “mil cuatrocientos noventa y dos” (“one thousand four hundred and ninety-two”).

- Names of months, days of the week and seasons: here it was necessary to check the POS tagging, since some of them, such as “abril” (“April”), “julio” (“July”) or “domingo” (“Sunday”), are also person names in Spanish.

- Granularities: Añotador distinguishes here between *DGRANULARITY* (anything bigger than a day, included, and that is considered DATE by the standard) and *TGRANULARITY* (anything smaller than a day, considered TIME by the standard), but for instance, in the case of DURATIONs (like “dos días”, meaning “two days”, and “una hora”, meaning “one hour”) they share common rules. Regarding the calculus, each granularity has information associated. For instance, both “siglos” or “centurias” (meaning both “centuries” in Spanish) are measured in years, and each one corresponds to 100 years. The concept century (stored as “100_YEARS” in the system) has therefore an associated granularity of years (“Y” regarding the standard) and an associated amount of 100, and it is identified when the lemmas “siglo” or “centuria” are used. If eventually we wanted to use another synonym, we would just have to add in the list of granularities of the rule file a new entry that maps to “100_YEARS”.

- Parts of the day or specific relative days: such as “tarde” (“afternoon”) or “ayer” (“yesterday”).

Añotador also detects other expressions, such as ordinals and roman numerals, and assigns them a value. All of them are tagged with basic annotations, such as numeric values and the type of expression, that will be used afterwards in other rules.

6.1.1.2 Basic temporal expressions

Once the most basic elements are identified, the next task is to combine them to detect temporal expressions. Some rules that can be found at this stage are shown in Table 6.1, where it can be seen how the result of some rules rely on previous annotations by other rules.

Apart from the results shown in the table, also other internal tags are stored, depending on the type of temporal expression:

- TIME: in the case of TIME, Añotador stores the hour, the minute, the second and the part of the day of the temporal expression.

- DATE: for DATEs, Añotador keeps the day, the week, the day of the week, the month and the year of the temporal expression.

- DURATION: for DURATIONs, Añotador stores the granularities in the temporal expression, such as the amount of years, days and hours.

#	Example	Pseudo-pattern	Tagged as	Value in the example
1	dos días *two days*	number + granularity	P + value number + value granularity	DURATION (P2D)
2	cada dos semanas *every two weeks*	cada + [DURATION]+	value of DURATION	SET (P2W)
3	a las tres *at three*	a? + las + [hour] + ![noun]	"T" + hour + ":00"	TIME (T3:00)
4	las tres menos 5 *five to three*	a? + las + [hour] + menos + [minutes]	"T" + (hour-1) + ":" + (60-minutes)	TIME (T2:55)
5	las tres de la tarde *three in the afternoon*	[TIME] [de|en] + [la|el] + [PARTDAY]	am/pm value of TIME	TIME (T15:00)
6	el 1 de Mayo de 1991 *May 1, 1991*	[day] + de + [month] + de + [year]	year + "-" + month + "-" + day	DATE (1991-05-01)

Table 6.1: Example of rules used to detect basic temporal expressions.

Although in most common temporal expressions these values are never used, sometimes we will find time expressions where part of the info is omitted in its extent (e.g. in "de uno a dos días", meaning "from one to two days", the first expression includes no granularity). Being able to retrieve it from close expressions will be useful; they are what in this thesis is called compound expressions.

6.1.1.3 Compound expressions

Compound expressions are time expressions where some information from one time expression must be used in another one for normalization (e.g., the previously mentioned expression "from one to two days"). To this aim, the information from the rules presented in the previous subsections will be used. Some examples of these types of rules can be found in Table 6.2.

6.1.1.4 Literal expressions

Apart from the previous rules, there are some token-based rules that target literal expressions, such as bank holidays or specific noun phrases. Some of these expressions are shown in Table 6.3. Please note that some of the expressions (such as #2 "el ayer" and #3 "el día de mañana") include time expressions with a different meaning ("ayer" means "yesterday", and "mañana", "tomorrow" or "morning"), so it is necessary to capture these expressions literally and avoid conflicts to the alternative interpretations. In fact,

#	Example	Pseudo-pattern	Tagged as	Value in the example
1	dos años, seis semanas y un día *two years, six weeks and one day*	[[DURATION]+ [,|y]]+ [DURATION]+	P+value of each duration	DURATION (P2Y6W1D)
2	el 1, el 2 y el 3 de mayo de 2011 *1st, 2nd and 3rd May 2011*	[*dayNum* [,|y]]+ [DATE]	each *dayNum* gets the info from DATE	DATE (2011-05-01, 2011-05-02, 2011-05-03)
3	mayo y junio de 2060 *May and June of 2060*	[*Month* [,|y]]+ [DATE]	each *Month* takes the year from the DATE	DATE (2060-05, 2060-06)
4	de uno a dos años from one to two years	[de|entre...] *num* [a|hasta...] [DURATION]	*num* inherits granularity from DURATION	DURATION (P1Y, P2Y)

Table 6.2: Example of compound rules.

the word "mañana" is especially tricky, since it can be used in several expressions in Spanish, some of them shown below:

- "mañana" (femenine noun) means "morning".
- "mañana" (adv) means "tomorrow".
- "pasado mañana" (adv) means "the day after tomorrow".
- "pasado" (adv) has the same meaning as "pasado mañana" (this is, "the day after tomorrow").
- "pasado" (noun or adjective) means "past" (noun or adjective).

Additionally, idioms and structures containing the term "mañana" change depending on the language variant. The expression "in the morning" is said with the Spanish of Spain "por la mañana", but "en la mañana" in other varieties in Latin America. Since most POS taggers fail to annotate these words in their variate senses, Añotador includes several rules (such as checking the presence of specific surrounding words) just to disambiguate them and maximize accuracy when dealing with these polysemic expressions.

The next section will detail how the normalization algorithm takes the result from the rules and transforms it into a valid value.

#	Temporal Expressions	Tagged as	Value
1	hoy en día, a día de hoy, en la actualidad *nowadays, currently*	DATE	PRESENT_REF
2	previamente, antaño, recientemente, el ayer, en el pasado *previosuly, recently, the past*	DATE	PAST_REF
3	el día de mañana, en los próximos años *in the next years*	DATE	FUTURE_REF
4	Nochevieja, Fin de Año *New Year Eve*	DATE	XXXX-12-31
5	Halloween	DATE	XXXX-10-31

Table 6.3: Some literal expressions.

6.1.2 Normalization Algorithm

Once the rules are applied, their results are processed through a normalization algorithm (this algorithm is explained step-by-step in the Annex B) that decides the final value of each expression. In the results, the value can come in different formats.

- Complete value: the value passed from the rules is already final.
- Incomplete value: value with Xs on them, usually dates. For instance, in *XXXX-02-22* we know the day and the month of the date, but not the year.
- Anchor functions: they are useful to represent dates with regard to another date.

While incomplete values are usually completed using the anchor date or the last full date found in a straightforward manner, in some cases the calculus is not that easy. For these more complex temporal expressions, anchor functions are used. An example of anchor would be that the value returned by the expression "Tomorrow" from the rules would be `anchor(TODAY,+,1D)`, meaning that the normalization value should be adding one day to the reference date.

For calendar calculus, Añotador relies on the widely-used library JodaTime[115], which supports basic operations such as adding months or days to a date or converting dates from different formats. Nevertheless, for more complicated operations, such as finding out the calendar date of a specific weekday (e.g., "el próximo lunes", meaning

[115] `https://www.joda.org/joda-time/`

"the next Monday", depends on the day of the week we are) or working with seasons, a set of specific functions that complement JodaTime utilities was developed.

6.1.2.1 Normalization of DATEs

DATEs are undoubtedly the most tricky temporal type. Despite some rules can output directly their final value for absolute dates (e.g., "6 de Marzo de 2019", meaning "6 March 2019", will return "2019-03-06"), most of them will need further normalization. Here the concept of *anchor date* (the date used as a reference for calendar calculations) becomes crucial. At the beginning of the processing, the anchor date will be the date provided to the system, but as the algorithm advances on the text, the last date found will be saved in order to be used as a reference date if needed. For instance, if we had the text "El 4 de julio estudió por la mañana, pero no por la tarde." ("July 4, he studied in the morning, but not in the afternoon."), we understand that the mentions to parts of the day ("morning" and "afternoon") do not refer to the present day, but to the previously mentioned date "July 4". Añotador would therefore normalize it to "El 4 de julio (2019-07-04) estudió por la mañana (2019-07-04TMO), pero no por la tarde (2019-07-04TAF).".

When a temporal expression (TE) is detected, the first step is to check if part of a DATEs is unknown (this is, the value returned by the rule includes "XXXX" or "XX"). If this is the case, it is normalized to the anchor date. This tends to happen when there are abstract mentions to days of the week or months (e.g, "en mayo", meaning "in May"). If this was not the case, there are two options: the value was absolute (so it is already the final value), or it is anchored. If the latter, we can find several types of anchoring, that are analyzed hereunder:

1. The TE refers to a previous or a future specific date, day of the week, weekend, month or season.

2. The TE refers to a specific granularity of the anchor date (e.g. "este mes", "this month").

3. The TE refers to a point in time resulting from adding or subtracting some duration from the anchor date (e.g. "ayer", meaning "yesterday", means to subtract one day from the present day).

The first case comprehends expressions such as “el verano pasado” (“last summer”), where the anchor date must be taken as a reference to decide to which date they refer to. These expressions should not be confused with others such as “el mes pasado” (“last month”). While this expression consists just of subtracting an amount of time from the anchor date (one month in the example), the ones targeted in this section require a bit more sophisticated normalization. If we say “last summer” in October 1991, we make reference to the summer of 1991, but also if we say it in March 1992. But in May 1991, we would refer probably to 1990. The same happens with days of the week, weekends, specific dates (e.g. “last 5th May”) or literal months (e.g. “next October”). To deal with these expressions, a set of functions that work over JodaTime on each specific granularity has been created.

The second case focuses on expressions such as “este mes” (“this month”), or “el año” (“the year”), where the time expression refers to some granularity of the anchor date. It is not always a value that can be directly extracted from the anchor date (such as the day, the month or the year), since it can also be a coarser granularity, such as the century the current date belongs to (e.g., “este siglo”, meaning “this century”). The rules of the system in this case return the desired granularity, and the normalization algorithm infers the correct normalized value from the anchor date.

The last case of anchoring implies adding or subtracting durations, such as in the expressions “yesterday” and “the day after tomorrow”[116]. In this case, the rules gather all the DURATIONs and express them as a single concatenation (e.g. “P3M2W1D” for “three months (M), two weeks (W) and a day (D) ago”). At this point, the algorithm iteratively uses JodaTime and the created functions to add or subtract each of them. So if the anchor case in the previous example was “2019-12-20”, the system would first subtract three months (2019-09-20), then two weeks (2019-09-06) and finally one day (2019-09-05), obtaining the desired date. The part of the algorithm doing these operations is disabled in the *Standard* option of Añotador for expressions such as “two days ago” or “in three months and two weeks” –working just for expressions like “tomorrow”– because of the TimeML guidelines, that specifically asks to annotate them as DURATIONS – but it can be re-activated if required, since it is useful for tasks such as timeline creation, as it was done for the CENDOJ implementation of the software.

[116]As explained in Section 6.1.1.4, “pasado” or “pasado mañana” is a particular expression in Spanish denoting “the day after tomorrow”.

Complementary expressions to this last case would be for instance “the rest of the year” (“lo que resta/queda de año”) or “the part of the month that already passed” (“lo que va/llevamos de mes”), where the result returned would be a composed duration (e.g., for the 2nd March the system would return “P2M1D”, two months and a day).

6.1.2.2 Normalization of other types of TE

Not only DATEs are normalized by the algorithm. The final value of a DURATION is also an output of the normalization algorithm. This process is similar to the parsing of DURATIONs introduced in the last case of the previous Section 6.1.2.1.

TIME expressions are also normalized by this algorithm. For instance, if we found a part of a day (e.g. “night”, normalized by the rules as “TNI”) or a time (e.g. “at 7 pm”), the system would anchor it to the current anchor date (e.g. “2019-12-20TNI” and “2019-12-20T19:00”, respectively).

Moreover, in the legal variant of the tool the INTERVAL detection is added. This functionality operates after normalizing all temporal expressions, and looks for patterns that surround them, such as “from X to Y”.

6.1.3 Availability

The source code and the rules of Añotador are available in GitHub[117] under a GNU GPL-3.0 license[118]. The code also includes methods to evaluate the tools’ performance against different corpora. It requires no external installation besides Maven dependencies.

There is also a visual demo[119] where the users can use the tool. The service can also be invoked as an HTTP rest service via cURL or Postman (a Postman collection of requests is also available to facilitate its use), receiving plain text and returning the annotations in TIMEX3 or JSON.

The pipeline of the tool is illustrated in Fig. 6.1. First, the text is processed using the CoreNLP pipeline (where IxaPipes’ models were added). In the TokensRegex annotator, the customized rules are used, and the input text is processed in different stages. Then these expressions are passed to the normalization algorithm, which processes them

[117]`https://github.com/mnavasloro/Annotador`
[118]`https://www.gnu.org/licenses/gpl-3.0.html`
[119]`http://annotador.oeg.fi.upm.es/`

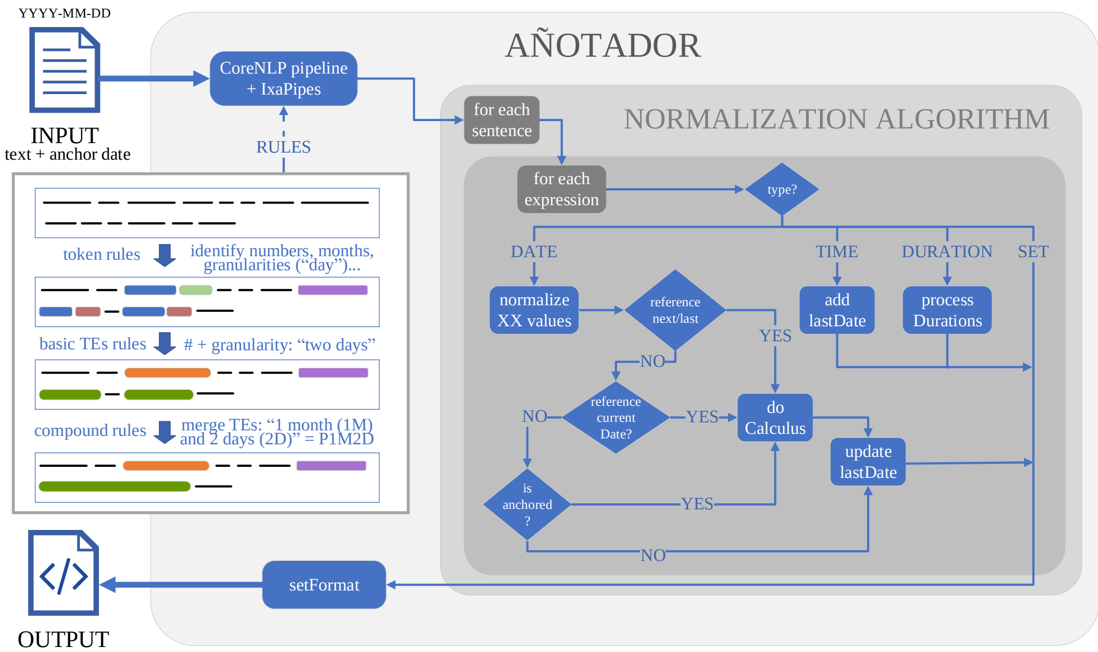


Figure 6.1: Pipeline of Añotador

differently depending on their type. In the case of DATEs, first, the unknown values are normalized, marked with X (e.g., "XXXX-05" means we know it is the month of May but we have no further info on the year, so it is normalized to the anchor date). Then we can have references to next or last points in the calendar (e.g., "last December"), references to the current date at some specific granularity ("this month") or anchorings where we have to add or subtract specific amounts of time (e.g. "one year and two months ago"). Once this processing is finished, the last date is stored (to know the value of possible anaphoras in the same sentence) and the system goes for the next expression. This is done for all the sentences in the text (at each sentence restarts the last date variable, used as anchor date within the sentence, to the original anchor date if different), and then the annotated text is returned in the required format (for now, TimeML TIMEX3 tags or JSON).

6.1.4 Special Cases

As reported in Chapter 3, both the legal domain and colloquial expressions present some particularities related to time that cannot be properly represented in the TimeML standard. In this section, how the approach of the thesis diverges a bit from the standard to better represent the needs of temporal annotation in general and the legal domain in particular has been presented. To do this, the following tasks were carried out:

1. Analysis of limitations of the standard, derived from the annotation of different corpora (further introduced in Chapter 5), namely (a) the HourGlass corpus, short texts in Spanish, where colloquial expressions were especially difficult to annotate following the standard, and (b) the TempCourt corpus, where several difficulties in the legal domain and lacks of the standard were detected and analyzed.

2. Feedback from domain experts: the following sources provided valuable feedback on the annotation needs:

   (a) Lynx partners: in the frame of the Lynx project, different industrial partners fulfilled a questionnaire about the temporal information they needed to be annotated.

   (b) Center of Judicial Documentation (CENDOJ, Centro de Documentación Judicial): CENDOJ experts also provided feedback on a preliminary annotation by the software Añotador, polished throughout different sessions.

Regarding the legal domain, Añotador complies with TimeML as far as possible, and included just new tags for intervals and a new option to normalize the granularity business days (`BD`). Additionally, some frequent temporal expressions that are not used in this sense, but are frequent taglines in the legal domain (such as "now"), were omitted. Also, some improvements for Spanish have been implemented framed in the collaboration with CENDOJ, such as normalizing expressions like "dentro de dos años" ("two years in the future") as the date it refers to, and not as a DURATION, that is the way the TimeML guidelines recommend.

In addition to the time-related particularities from the legal domain, also other considerations were taken into account. One of them is the format; legal texts have very specific formats depending on the type of document and the organism that produced them. For this reason, Añotador had to be able to process different formats such as XML or HTML and merge them to the output, since legal documents tend to include a lot of metadata or tags. One of the requisites when dealing with documents from CENDOJ was precisely to be able to take into account the tags already in the documents, both for distinguishing the different parts of the document and for dealing with specific markups, such as links or citations.

Regarding Spanish, Añotador covers cases that state-of-the-art temporal taggers do not meet. One of the clearest examples is the case of the word *mañana*, really frequent in Spanish, already presented in Section 6.1.1.4. The system also covers different registers, being for instance able to detect expressions like *antaño*, a cultured way to say "in the past" not frequent in news, as well as some colloquial expressions. Additionally, Añotador also covers some Latin American expressions, that as far as the doctoral candidate knows, have never been considered in previous temporal taggers, despite the Latin American Spanish being much more spoken than Castilian Spanish.

Regarding the legal domain, the fact that most temporal taggers were not able to identify years when written with letters, in particular for Spanish, was noticed. This is an extremely important feature, especially due to legal documents in Spanish, like contracts, BOE documents (from the Spanish National Gazette) or judgments, tending to express years in this format. Finally, also composed DURATIONs (e.g. "one day and three hours") are often annotated separately by previous temporal taggers, so measures to correctly address this issue were taken.

6.1.5 Use cases

Añotador has been successfully used in different domains and languages. The main use cases are detailed below.

6.1.5.1 Use Case 1: Lynx Project

Lynx[120] is an H2020 EU project aiming to build a Legal Knowledge Graph for Smart Compliance Services in Multilingual Europe in order to help small and medium-sized enterprises to deal with multilingual compliance. It comprises different use cases and a portal that makes use of different services (summarization, search, etc) that rely on different independent microservices. One of these microservices tackles temporal tagging and uses Añotador for the processing of texts in Spanish and English.

In the context of this project, Añotador deals with many different kinds of legal texts, and it has been forced to adopt different deployment methods. As a result,

[120] `https://lynx-project.eu/`

Añotador has been integrated into a grid of microservices and is successfully requested for different uses in the domain, as has been documented in different deliverables[121]. Additionally, the questionnaire in Annex A was distributed in order to retrieve temporal tagging related requests.

6.1.5.2 Use Case 2: Collaboration with CENDOJ

During this thesis, the PhD candidate had the opportunity to collaborate with CENDOJ. Añotador was tested against some of their documents, confronting its legal mode to some real-world cases (the permission to do so is attached in Annex C). Since the documents and the specific software of this use case are not public, it will be explained briefly, in a shallow way.

Although the idea was to create a corpus of judgments annotated with events, due to limitations beyond our control this was changed to an iterative evaluation, that proceeded as follows:

1. First, a few judgments from CENDOJ were annotated by Añotador and sent back to CENDOJ for evaluation.

2. Then, these annotations were manually checked and a detailed report of the errors was sent back. These errors were discussed in a call, where the expected correct results were detailed, together with some particularities from the language used that had to be taken into account.

3. The failures detected were solved, and new functionalities were added to Añotador in order to heed the considerations previously discussed. Some of them include considering differently dates in links and taking into account the structure of the document. To this aim, a code to deeply process the structure of CENDOJ documents was developed.

4. A new bench of documents was then annotated and sent back in order to get feedback. A new evaluation was returned, and Añotador was again refined.

5. New documents were then sent back in order to be annotated, this time also with events. Regrettably, this step could not be accomplished.

[121]`https://zenodo.org/record/3235752`, `https://doi.org/10.5281/zenodo.3865668`

Some of the main changes resulting from the feedback provided by CENDOJ are summarized below:

- Many references to times were not correctly processed, because they are similar to mentions of durations (e.g. "*A las 23 horas*", meaning "*At 11pm*").
- Intervals were added, since many ranges are mentioned within the judgments and they are regarded as important by CENDOJ experts.
- References to the present time without a real temporal intention are no longer annotated in legal texts.
- Some uses of capital letters and money symbols were misleading to the system, so Añotador was made more robust in order to deal with them.

6.1.5.3 Use Case 3: As an occasional service within other tasks

Finally, Añotador has been also used by other services in order to accomplish different tasks. Some examples are detailed below:

Terminology Extraction In order to retrieve terminology from texts, TermitUp[122] uses Añotador to avoid dates to be included as relevant words.

Translation Memory Matching A recent work on Translation Memory Matching and Retrieval (Ranasinghe et al., 2020) used Añotador together with other NLP tools for Spanish in order to detect dates and named entities.

Anonimization Finally, Añotador was also used in a national project in order to delete dates from texts in order to have them anonymized.

6.1.6 Evaluation

This section will present the evaluation of Añotador both from the user validation point of view (Section 6.1.6.1) and testing it against corpora (Section 6.1.6.2).

[122] `https://termitup.oeg.fi.upm.es/`

6.1.6.1 User validation

In the legal domain, the tool has been used in the context of the Lynx project for both English and Spanish. Additionally, the CENDOJ use case significantly refined the annotation of Spanish sentences. Both use cases were described in Section 6.1.5. Additionally, Añotador has been used as an API for several NLP tasks and no problem has been reported.

Also, general users were asked to test the tool and report the main problems they found. Most of the comments were tackled, and they were gathered in a spreadsheet[123] to keep track of their implementation. For this evaluation, they used the demo of Añotador available on its webpage, which is freely accessible to any user.

6.1.6.2 Corpora Evaluation

In this section, the results of the created temporal tagger against different corpora will be presented. As explained in Section 4.7, different aspects of temporal expressions will be covered, namely extension identification, normalization of the value and type. The evaluation has been performed in terms of *precision* (this is, the share of hits among the expressions tagged by the tools), *recall* (the share of hits among all the expressions to be tagged) and *F1-measure* (the average of *precision* and *recall*). These metrics will be considered *lenient* (this is, a partially tagged expression is considered a hit, even if not all its extent is marked by the tagger), *strict* (just expressions tagged exactly as in the test are considered correct) and *average* (average of *lenient* and *strict*). In order to extract these metrics from the results of the taggers, the software GATE (Cunningham et al., 2013) was used.

General Spanish

In this section, the results of Añotador against HeidelTime and SUTime for General Spanish are presented, since there are no public legal corpora available in Spanish and the legal suitability of the software has already been validated for Spanish by CENDOJ.

For the evaluation, HeidelTime was called using the following parameters: "News" as the type of text, "Spanish" as language and "TreeTagger" as POS tagger. On the other

[123] `https://docs.google.com/spreadsheets/d/1jv5bcsPs9YaitoN9IAP3kHwFzdZxtVDyzkbuANec094/edit?usp=sharing`

hand, SUTime was invoked directly, not via the NER Annotator, as in the example code available in its documentation[124], but using the Spanish properties. Although it was also intended to evaluate the temporal tagger TIPSem running it on different machines and configurations, it was not possible to use it to process Spanish texts (despite it working fine for English) due to the unavailability of some auxiliary software required by TIPSem[125]. The same attributes as in the TempEval 2 challenge were considered for evaluation, (1) the identification the extent of the TE (*extent*), (2) the identification of the type of TE (*type*) and (3) the normalization (*value*). Apart from the HourGlass corpus, also the TempEval 2 corpus was used for evaluation[126].

HourGlass corpus Table 6.4 shows the result of Añotador, SUTime and HeidelTime (with the configuration detailed in Section 5.2.3.1) on the HourGlass corpus (the complete output can be found in the website of the corpus).

Despite Añotador getting the best results, all the taggers shared some common errors, such as not tagging expressions such as in the document named *90061*, from the people part of the corpus, with the text "Era a las 19 o a y 15?" ("Was it at 19 or at (19:)15?"), where none of the taggers was able to identify these expressions written in a colloquial way.

For instance, none of them found the colloquial expression "en cero coma", that means "in seconds" (doc 90001). In doc 90065, not HeidelTime nor SUTime found the expression "lo vuestro dura 1h, no?" ("your stuff lasts 1h, right?"; Añotador correctly marked it, but wrongly considered it a TIME instead of a DURATION. Similarly, in the case of compound durations (such as "1 año, 6 meses y un día", "1 year, 6 months and one day", from doc 00060), each tagger performed differently: Añotador correctly marked it all as a full expression, HeidelTime tagged each part individually and SUTime recognized no time expression. Finally, both HeidelTime and SUTime have problems when

[124] `https://nlp.stanford.edu/software/sutime.shtml`

[125] Nevertheless, the PhD candidate would want to thank its creator for his support and help during the process

[126] As explained in Section 2.2.1, Spanish corpora are really scarce: TempEval 3 test dataset is not available, TimeBank ModeS is for old Spanish, TimeBank has the same documents as TempEval 2 and 3 and MEANTIME corpus does not annotate all the time expressions in a document, so it was not suitable. Also, it must be noted that the scorer available for TempEval 2 did not include the key documents for Spanish, and did not work. The key documents were therefore obtained from GitHub (`https://github.com/AntonFagerberg/Temporal-Information-Extraction/tree/master/tempeval2-data`).

		strict			lenient			average		
Tagger	Attribute	P	R	F1	P	R	F1	P	R	F1
Añotador	value	0.72	0.71	0.72	0.80	0.78	0.79	0.76	0.74	0.75
	type	0.79	0.77	0.78	0.89	0.87	0.88	0.84	0.82	0.83
	extent	0.83	0.82	0.82	0.95	0.92	0.94	0.89	0.87	0.88
Heidel	value	0.57	0.48	0.52	0.64	0.53	0.58	0.60	0.51	0.55
	type	0.61	0.51	0.55	0.82	0.69	0.75	0.72	0.60	0.65
	extent	0.62	0.52	0.57	0.87	0.73	0.80	0.75	0.63	0.68
SUTime	value	0.30	0.08	0.13	0.45	0.12	0.19	0.38	0.10	0.16
	type	0.47	0.13	0.20	0.80	0.21	0.34	0.64	0.17	0.27
	extent	0.47	0.13	0.20	0.89	0.24	0.37	0.68	0.18	0.29

Table 6.4: Results of the temporal taggers in the HourGlass corpus. Añotador has the highest results, although HeidelTime also shows good performance. All the taggers show worse performance in comparison to the TempEval 2 corpus, although the difference is smaller in the case of Añotador.

recognizing literal numbers in Spanish – such as in doc 00011, where in "En el año mil" ("In the year one thousand.") HeidelTime just recognized "year one" and SUTime tagged nothing. This problem also appears when dealing with polysemic expressions such as "pasado" and "mañana" (previously introduced in Section 6.1.1.4); doc 00008 includes the text "Ya lo veremos pasado mañana." ("We will see it the day after tomorrow."), where SUTime just recognizes "mañana" as "tomorrow" and HeidelTime tags the expression correctly but considers that it refers to the morning of the previous day. Regarding Latin American Spanish, only Añotador recognizes expressions like "Cinco para las 11." ("Five to eleven.", doc 90053).

TempEval 2 The TempEval 2 corpus has 175 documents for training and 35 documents for testing. Since the documents were in the .tab format but the Python scorer facilitated in the website did not work, these tab files were transformed into plain text documents for testing the output of the temporal taggers using GATE.

In Table 6.5 the results obtained by Añotador, HeidelTime and SUTime are shown. As in the previous evaluation, SUTime precision is generally its highest metric, although it is in most cases beaten by Añotador and HeidelTime. On the other hand, Añotador's recall is the highest in all cases. Regarding F1-measure, Añotador tends to be better for detecting the *extent* of the tag and its *type*, while HeidelTime is slightly better on normalizing the *value*. Overall, Añotador is better than HeidelTime in most of the metrics, having similar results when not, and both tend to surpass SUTime.

		strict			lenient			average		
Tagger	Attribute	P	R	F1	P	R	F1	P	R	F1
Añotador	value	0.80	**0.78**	0.79	0.83	**0.80**	0.82	0.82	**0.79**	0.80
	type	0.84	**0.82**	**0.83**	0.91	**0.88**	**0.89**	**0.88**	**0.85**	**0.86**
	extent	0.87	**0.84**	0.85	0.93	**0.90**	**0.92**	0.90	**0.87**	**0.89**
Heidel	value	**0.84**	0.75	**0.80**	**0.86**	0.77	**0.82**	**0.85**	0.76	**0.81**
	type	**0.85**	0.76	0.81	0.89	0.79	0.84	0.87	0.78	0.82
	extent	**0.90**	0.81	**0.85**	0.94	0.84	0.89	**0.92**	0.83	0.87
SUTime	value	0.64	0.22	0.33	0.83	0.29	0.43	0.73	0.26	0.38
	type	0.65	0.23	0.34	**0.93**	0.32	0.48	0.79	0.28	0.41
	extent	0.67	0.23	0.35	**0.96**	0.33	0.49	0.81	0.28	0.42

Table 6.5: Results of the temporal taggers in the TempEval 2 corpus –best metrics are highlighted in bold. Although HeidelTime is slightly better at finding the normalized value (0.0027 on average), Añotador is better in the rest of metrics. SUTime obtains high precision but low recall.

Legal English

For the English language, the focus is mainly on covering legal texts. In order to test the system, Añotador is used against the TempCourt corpus (described in Section 5.1). Table 6.6 shows the results of Añotador on the ECHR part of the corpus in comparison to ten state-of-the-art temporal taggers whose annotations are included as a benchmark (Section 5.1.3), while Table 6.7 and Table 6.8 do so for decisions from the ECJ and the USSC courts, respectively.

Table 6.6 shows the good performance of most taggers on the ECHR subcorpus, since they tend to find the same number of annotations that appear in the gold standard, especially if we focus on the *lenient* figures, showing that the errors are mostly in the extension of the tagging more than in its identification. In the ECJ and USSC subcorpora (Tables 6.7 and 6.8 respectively), where the texts are more complex in general, the number of annotations by the taggers differs from the gold standards. One of the reasons for this is, in the case of the ECJ section of the corpus, that the designators of European legal acts such as regulations and directives follow a special scheme which

	lenient			strict			lenient + value			strict + value		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
AÑ	0.98	0.96	0.97	0.94	**0.93**	**0.93**	**0.91**	**0.89**	**0.90**	**0.88**	**0.87**	**0.87**
	0.87	0.97	0.92	**0.83**	**0.93**	**0.88**	**0.81**	**0.90**	**0.85**	**0.77**	**0.86**	**0.81**
HE	**0.99**	**0.99**	**0.99**	0.84	0.84	0.84	0.78	0.78	0.78	0.78	0.78	0.78
	0.88	**0.99**	**0.93**	0.71	0.80	0.75	0.67	0.75	0.71	0.64	0.72	0.68
SU	0.88	0.87	0.88	0.85	0.84	0.84	0.78	0.78	0.78	0.76	0.75	0.75
	0.76	0.85	0.80	0.71	0.80	0.76	0.66	0.74	0.79	0.64	0.72	0.68
GU	0.96	0.93	0.94	**0.95**	0.92	**0.93**	0.86	0.84	0.85	0.86	0.84	0.85
	0.84	0.92	0.88	**0.83**	0.92	0.87	0.74	0.82	0.78	0.74	0.82	0.78
CA	0.88	0.87	0.87	0.83	0.82	0.82	0.78	0.78	0.78	0.75	0.75	0.75
	0.75	0.85	0.80	0.70	0.79	0.74	0.65	0.74	0.69	0.64	0.72	0.67
CL	0.92	0.78	0.85	0.34	0.32	0.35	-	-	-	-	-	-
	0.80	0.77	0.78	0.33	0.32	0.33	-	-	-	-	-	-
SY	0.98	0.93	0.96	0.83	0.79	0.81	0	0	0	0	0	0
	0.86	0.93	0.90	0.70	0.76	0.73	0	0	0	0	0	0
TE	0.94	0.95	0.95	0.92	**0.93**	0.92	0.86	0.88	0.87	0.85	0.86	0.85
	0.83	0.95	0.89	0.80	0.92	0.85	0.75	0.86	0.80	0.72	0.83	0.77
TI	0.78	0.85	0.81	0.64	0.70	0.67	0.64	0.71	0.67	0.63	0.69	0.66
	0.69	0.86	0.76	0.62	0.77	0.69	0.64	0.79	0.71	0.61	0.76	0.68
US	0.73	0.61	0.67	0.69	0.58	0.63	0	0	0	0	0	0
	0.65	0.62	0.64	0.61	0.58	0.60	0	0	0	0	0	0
UW	0.90	0.53	0.67	0.51	0.30	0.38	0.55	0.33	0.41	0.51	0.30	0.38
	0.86	0.58	0.69	0.48	0.32	0.38	0.51	0.34	0.41	0.48	0.32	0.38

Table 6.6: Evaluation results for the ECHR corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) correspond to results against the *Standard-TimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

also includes the year when the legal act has been agreed, such as ***2016**/679*. Despite this is not a temporal reference, some taggers find it misleading.

If we briefly consider the results on each corpus, we can see that in the ECHR corpus most taggers perform equally well when *strictly* evaluated, while GUTime provides the best results after Añotador, closely followed by TERNIP. On the contrary, TIPSem, USFD2 and UWTime are not as performant. This is because the ECHR uses fully qualified dates (e.g. *10 January 2017*) and does not include many references to other court decisions. On the other hand, the ECJ corpus results present one outlier that can be spotted immediately: it is the precision of the HeidelTime annotations, that is significantly different from its other precision values across each section of the corpus. Finally, the USSC corpus is slightly different to ECHR and ECJ, containing American English and using a different date format. It also repeats part of the text in the judgment, which leads to poorer performance as incorrect annotations are also repeated.

	lenient			**strict**			**lenient + value**			**strict + value**		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
AÑ	**0.98**	0.94	**0.96**	0.96	0.92	**0.94**	**0.96**	0.92	**0.94**	0.94	0.90	**0.92**
	0.54	0.94	**0.69**	0.52	0.91	**0.67**	**0.53**	0.92	**0.67**	0.51	0.89	**0.65**
HE	0.48	0.95	0.64	0.47	0.94	0.63	0.47	**0.94**	0.62	0.47	**0.93**	0.62
	0.27	0.97	0.42	0.26	**0.96**	0.41	0.26	**0.94**	0.40	0.26	**0.93**	0.40
SU	0.81	0.97	0.88	0.79	**0.95**	0.86	0.78	0.93	0.85	0.77	0.92	0.84
	0.44	0.95	0.60	0.43	0.93	0.58	0.41	0.90	0.57	0.41	0.89	0.56
GU	**0.97**	0.87	0.91	**0.97**	0.86	**0.91**	0.94	0.84	0.89	0.94	0.84	0.88
	0.51	0.82	0.63	0.50	0.82	0.62	0.48	0.78	0.60	0.48	0.78	0.60
CA	0.89	0.74	0.81	0.85	0.70	0.77	0.86	0.71	0.77	0.85	0.70	0.77
	0.49	0.74	0.59	0.46	0.70	0.56	0.46	0.70	0.56	0.46	0.69	0.55
CL	0.77	0.88	0.82	0.32	0.36	0.34	-	-	-	-	-	-
	0.42	0.88	0.57	0.18	0.37	0.24	-	-	-	-	-	-
SY	0.89	**0.99**	0.93	0.81	0.90	0.85	0	0	0	0	0	0
	0.49	**0.98**	0.65	0.46	0.92	0.61	0	0	0	0	0	0
TE	0.97	0.88	0.92	0.96	0.88	0.91	**0.96**	0.87	0.91	**0.95**	0.87	0.91
	0.54	0.89	0.67	**0.53**	0.88	0.66	**0.53**	0.88	0.65	**0.52**	0.87	**0.65**
TI	0.72	0.81	0.76	0.64	0.72	0.68	0.62	0.70	0.65	0.61	0.69	0.65
	0.41	0.83	0.54	0.37	0.75	0.49	0.35	0.71	0.47	0.34	0.70	0.46
US	0.31	0.54	0.39	0.29	0.51	0.37	0.02	0.04	0.03	0.02	0.03	0.02
	0.20	0.65	0.31	0.19	0.61	0.29	0.02	0.06	0.03	0.02	0.05	0.02
UW	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-

Table 6.7: Evaluation results for the ECJ corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) correspond to results against the *StandardTimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

Regarding different date formats, they are a common challenge that emerges when applying temporal taggers to a corpus. Typically dates found across all evaluated documents are fully qualified dates containing a day, the month in full and a year. The formats in which these dates are provided are different for European and American

	lenient			**strict**			**lenient + value**			**strict + value**		
A	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
AÑ	0.74	0.73	0.74	0.56	0.56	0.65	0.66	0.65	0.66	0.53	0.53	0.53
	0.26	0.79	0.40	0.18	0.54	0.27	0.20	0.60	0.30	0.15	0.45	0.22
HE	0.83	0.94	0.88	0.81	0.92	**0.86**	0.79	**0.90**	**0.84**	0.79	**0.89**	**0.83**
	0.29	0.97	0.44	0.26	0.88	0.40	0.20	0.67	0.31	0.19	0.64	0.29
SU	0.75	**0.99**	0.85	0.72	**0.95**	0.82	0.67	0.88	0.76	0.66	0.86	0.75
	0.25	**0.98**	0.40	0.23	**0.90**	0.36	0.18	0.72	0.29	0.17	**0.69**	0.28
GU	0.84	0.78	0.81	0.71	0.66	0.69	0.67	0.62	0.65	0.65	0.60	0.62
	0.25	0.69	0.36	0.16	0.45	0.23	0.12	0.34	0.18	0.10	0.27	0.14
CA	0.77	0.90	0.82	0.72	0.84	0.77	0.73	0.85	0.78	0.71	0.83	0.76
	0.23	0.82	0.36	0.21	0.72	0.32	0.21	**0.73**	0.33	0.20	**0.69**	0.30
CL	0.85	0.84	0.84	0.81	0.79	0.80	-	-	-	-	-	-
	0.30	0.89	0.45	0.26	0.78	0.39	-	-	-	-	-	-
SY	0.85	0.98	**0.91**	0.78	0.91	0.84	0	0	0	0	0	0
	0.28	**0.98**	0.44	0.24	0.84	0.37	0	0	0	0	0	0
TE	**0.93**	0.86	0.90	**0.90**	0.83	**0.86**	**0.86**	0.79	0.83	**0.85**	0.78	0.81
	0.32	0.90	**0.48**	**0.29**	0.81	**0.43**	**0.25**	0.69	**0.37**	**0.23**	0.64	**0.34**
TI	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-
US	0.50	0.30	0.37	0.39	0.23	0.29	0.08	0.02	0.08	0.02	0.01	0.02
	0.16	0.28	0.21	0.07	0.13	0.09	0.08	0.14	0.10	0.03	0.05	0.04
UW	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-

Table 6.8: Evaluation results for the USSC corpus for each temporal tagger, both for identification (two first columns, *lenient* and *strict*) and normalization (two last columns, *lenient* and *strict*). The first row (in white) correspond to results against the *Standard-TimeML* gold standard, while the second (in gray) corresponds to the *LegalTimeML* gold standard.

sources of legal documents. The date in Europe is usually indicated in the format "Day Month Year" (e.g. *10 January 2017*), whereas the American date format is "Month DD, YYYY" (e.g. *January 10, 2017*). This particular difference in the date format has been processed correctly by some taggers, such as HeidelTime and SUTime, annotating both versions as a single date. GUTime however was not reliable in this context, even though it is the best tagger in the other corpora. It either detected only one part of the American-formatted date (e.g. *January 10*) or it treated both parts of the same date as two different annotations. The situation can also happen when dealing with short dates normalization (e.g., "10/05/2017" can mean "10 May 2017" or "5 October 2017", depending on if we consider the format to be “DD/MM/YYYY” or “MM/DD/YYY”).

Although the figures of some of the temporal taggers might seem unexpectedly high, considering the lack of domain adoption, it must be taken into account that they tend to be nevertheless less performant than results previously reported by taggers in general evaluations[127] (Chang and Manning, 2012).

In summary, although the evaluation results are promising it is worth noting that legal documents, especially court decisions, have some particularities (such as those highlighted in Section 3) which cause some stumbling blocks for automatic temporal taggers being applied out-of-the-box.

With regard to the comparison between the two reference standards *Standard-TimeML* and *LegalTimeML*), if we check the differences between figures and focus on the recall (since the taggers are not trained for the particularities of this annotation set, the precision is obviously not expected to be high and does not indicate the tagger’s usefulness), we see that the best taggers remain to be the same ones.

6.2 Additional Tool: lawORdate

In order to cover the gap related to legal references looking like dates, at the beginning of the thesis a tool to detect them was created. In this section this tool will be briefly introduced; more information can be found in the related publication (Navas-Loro, 2017).

lawORdate is a tool that cleans legal references with a date form from text documents. It addresses an important problem when processing legal documents from the temporal perspective, since common legal references in Spanish tend to include dates or patterns that can be misleading to temporal taggers. Although especially useful for legal texts, this tool is not just suitable for them, but also for any text including legal references in Spanish. Open data portals, for instance, also present legal references along with dates, and the first version of the tool was precisely designed to cope with this. The aim of the system[128] was to extract temporal coverage from both news and related datasets in Spanish, some of them in the legal domain, and be able to link them based on the temporal dimension. This system calls a temporal tagger (in the demo,

[127] `https://github.com/HeidelTime/heideltime/wiki/Evaluation-Results`
[128] `https://github.com/mnavasloro/AportaCuando`

HeidelTime (Strötgen and Gertz, 2013)), able to detect temporal expressions in texts in Spanish and tag them following the TimeML annotation standard. Nevertheless, this tagger happened to tag as temporal expressions references to Spanish laws and legal documents that led to false positives, such as shown in the example exposed in Fig. 6.2, extracted from a real article[129]. The result of the tagging by HeidelTime can be found in Fig. 6.3.

Estas actividades están reguladas por **Real Decreto 1341/2007, de 11 de octubre** *sobre la gestión de la calidad de las aguas de baño, incorporando al derecho español* **la Directiva 2006/7/CE del Parlamento Europeo y del Consejo de 15 de febrero de 2006** *relativa a la gestión de la calidad de las aguas de baño.*

Figure 6.2: Example of legal reference in a text. For English: 'These activities are regulated by ***Royal Decree*** **1341/2007, of 11th October** on the management of bathing water quality, incorporating into **Spanish law Directive 2006/7/ EC of the European Parliament and of the Council of 15th February 2006** on to the management of the quality of bathing waters.'

Estas actividades están reguladas por **Real Decreto** <TIMEX3 tid="t2" type= "DATE" value="1341">**1341**</TIMEX3>/<TIMEX3 tid="t3" type="DATE" value="2007">**2007**</TIMEX3>, <TIMEX3 tid="t9" type="DATE" value= "2016-10-11">**de 11 de octubre**</TIMEX3> sobre la gestión de la calidad de las aguas de baño, incorporando al derecho español **la Directiva** <TIMEX3 tid="t4" type="DATE" value="2006">**2006**</TIMEX3>**/7/CE del Parlamento Europeo y del Consejo** <TIMEX3 tid="t8" type="DATE" value="2006-02-15">**de 15 de febrero de 2006**</TIMEX3> relativa a la gestión de la calidad de las aguas de baño.

Figure 6.3: Result of Heideltime tagging. In blue, result of HeidelTime tagging on the text in Fig. 6.2.

This problem can also be found in the description of datasets, being especially problematic when obtaining obviously inconsistent dates such as happens in the example in Fig.6.4, extracted from the description of a real dataset[130]. The tagged dates without a legal-focused preprocessing were *'2093'*, *'2008'* and *'2008-12-19'*. While the latest

[129] `http://www.castillalamancha.es/actualidad/notasdeprensa/castilla-la-mancha-cuenta-con-35-zonas-de-ba%C3%B1o-autorizadas-donde-disfrutar-de-la-naturaleza`

[130] `http://datos.gob.es/catalogo/e04990501-registro-de-centros-tecnologicos-y-centros-de-apoyo-a-la-innovacion-tecnologica`. Last visited on 2017.

can at least be used as a lower temporal bound (since there is no additional temporal information on the coverage in the description), the year 2093 is obviously inconsistent.

Base de datos que proporciona información sobre los Centros Tecnológicos y Centros de apoyo a la Innovación inscritos en el registro creado mediante ***el Real Decreto 2093/2008, de 19 de diciembre****. Permite la consulta por Modalidad, Área Tecnológica, Sector, Comunidad Autónoma y/o Provincia. Además, posibilita la descarga de la versión completa en PDF.*

Figure 6.4: Example of text. For English: 'Database that provides information on Technology Centers and Innovation Support Centers registered in the registry created by ***the Royal Decree*** **2093/2008, of December 19**. It allows consultation by Modality, Technological Area, Sector, Autonomous Community and/or Province, as well as to download the full version in PDF.'

The aim of the web service lawORdate described in this section is to detect common legal expressions that tend to mislead temporal taggers and replace them in the text, in order to obtain a clean version of it where temporal taggers are able to detect only temporal expressions.

6.2.1 References detected

In the frame of news and dataset description processing, namely trying to locate them into a temporal instant or interval, several legal references happened to be tagged as temporal expressions by a state-of-the-art temporal tagger. Some examples are the following expressions, that refer to different official Spanish documents or laws:

- Ley Orgánica 10/1995 (*Organic Law).*
- Ley 22/2011, de 28 de julio (*Law).*
- BOE: 29/07/2011 or BOE de 22 de julio or BOE núm. 306, de 23 de diciembre (BOE: Boletín Oficial del Estado *Official State Gazette*).
- Real Decreto 1341/2007 (sometimes also expressed as RD 1463/2007, *Royal Decree*)
- Directiva 2012/27/UE.

These references are often also surrounded by a date referred to their creation (being therefore important to detect them as well). These legal expressions can also include additional words such as in “Real Decreto Legislativo” (“Legislative Royal Decree”) or be combined such as in “Real Decreto Legislativo 1/2004 de 5 de enero BOE de 8 de marzo”. Also, exceptions where dates near to references to legal documents can be found, such as happens when the dataset contains information about the proper legal document, such as in the example[131] depicted below, where the dates refer indeed to temporal coverage:

Publicaciones en Boletín Oficial del Estado (BOE): 2013-2017. (for English: *‘Publications in the Official Spanish National Gazette (BOE): 2013-2017.’*)

The problem of detecting these references is therefore not straightforward. In the following section, the method to identify some patterns for references found in a concrete application case (dataset descriptions and news) is detailed.

6.2.2 Detection of patterns

The corpus used is a dump of metadata from the Spanish open data portal datos.gob.es[132], consisting of almost 16k datasets. Some of them contained temporal coverage information expressed as *dcat:temporal property*[133], but most of them had their upload and creation date as only temporal information, along with information on the publisher, the title and the description.

A first analysis performed on metadata from these datasets showed that most appearances followed constrained patterns, and that texts that presented this kind of references misled the temporal tagger. Besides detecting temporal expressions that are not actually from the text timeline, another major problem derives from this pitfall: temporal normalization is also affected, since some dates can be wrongly normalized because of the misidentification of legal references nearby as temporal expressions.

Once these patterns are detected, they are replaced in the text by strings containing information of the legal references detected, but in a format that does not mislead

[131] `http://datos.gob.es/catalogo/l01280148-publicaciones-en-boletin-oficial-del-estado-boe-2013-2017`

[132] `http://datos.gob.es/`

[133] `https://www.w3.org/TR/vocab-dcat/#Property:dataset_temporal`

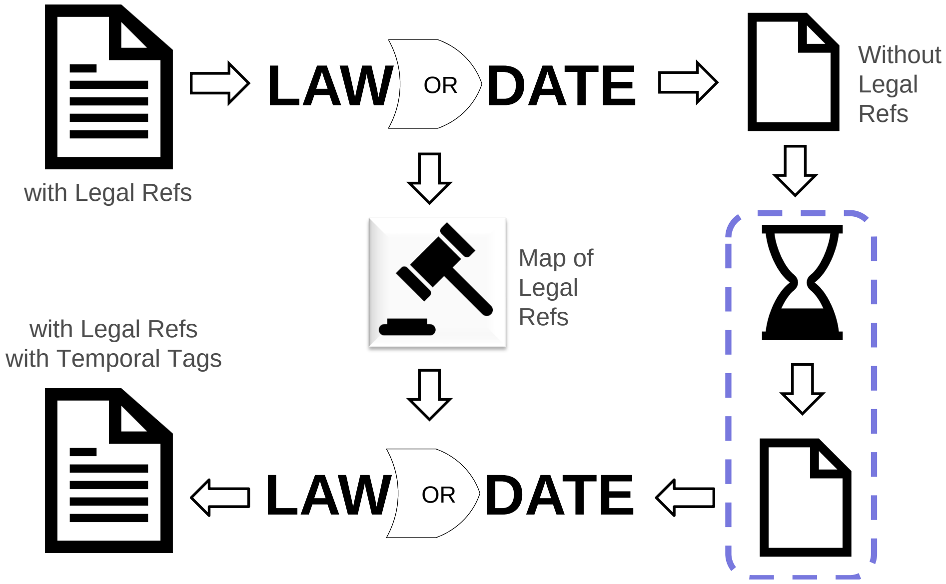


Figure 6.5: Pipeline of use of lawORdate.

the temporal tagger. This new version of the text maintains all the original genuine temporal expressions, being, therefore, the ones remaining those that must be detected by the tagger. Once the text is correctly tagged, old legal references can be recovered. Besides facilitating single-use temporal processing of isolated documents, this service also allows generating correctly temporally tagged texts with legal references that can be used for training machine-learning-based temporal taggers in order to adapt them to the legal domain.

lawORdate is currently available as a web application[134], and its source code is in GitHub repository[135], and finds and replaces misleading legal references in the texts, storing the original references. Once the temporal tagging is done, the references are restored in the text.

In the pipeline of use of lawORdate shown in Fig. 6.5, a text with legal references is first sent to the service. Then it finds all the misleading legal references that could affect the precision of a temporal tagger and replace them with innocuous expressions, storing the original references for further restoration. The output of this first step is to be used in a temporal tagger (in the demo, HeidelTime is offered, but any other can be used). Then, the output of the tagger (in TimeML) is sent back to lawORdate, which

[134] `http://legalwhen.appspot.com/`
[135] `https://github.com/mnavasloro/lawORdate`

restores the original legal references. The original text is therefore obtained, but tagged without the interference of any legal references in it. An example of use can be found in Figure 6.6.

6.2.3 Conclusions

This section presented shows how a basic preprocessing for detecting legal expressions to prevent temporal taggers from tagging them can improve temporal tagging on all kinds of legal-related texts. Also, other languages or kinds of texts could benefit from this preprocessing: the work made for Spanish and general but legal-related texts (news and datasets in the original case) can be adopted also for other languages, as it has been done for English in the processing of legal events in European decisions (see Section 8.2).

6.3 Challenges detected

In addition to the gaps covered by Añotador and the good figures of the temporal taggers, there still are open issues in temporal tagging. *Context-free TE* refer to fixed instants or intervals of time irrespective of any other consideration, and are already covered by most temporal taggers. *Context-dependent TE* (CDTEs), on the other hand, refer to precise instants or intervals, but in order to determine them, some additional context information is necessary. This context information can be present in the text in one form or another, from very explicit mentions to indirect hints from which it can be inferred. In the worst case, the context information will be tacit knowledge, shared only between the writer and a specific reader or reader type. We can identify here different types of CDTEs, where context information is necessary to determine (i) whether a group of words is a TE or (ii) how to normalize the TEs.

TE dependent on temporal information Whereas Añotador considers the anchor date as a date of reference to resolve relative references (e.g. "tomorrow"), the temporal information to be considered to disambiguate can be more complex. In the sentence: "Entre el golpe de estado del 18 de brumario y el 3 de nivoso." ("Between coup of 18 Brumaire and 3 Nivôse."), "nivoso" has two senses that need to be disambiguated by temporally framing the text. Besides the French Republican calendar, also other calendars have named historical facts, such as the Julian calendar and the October Revolution, that

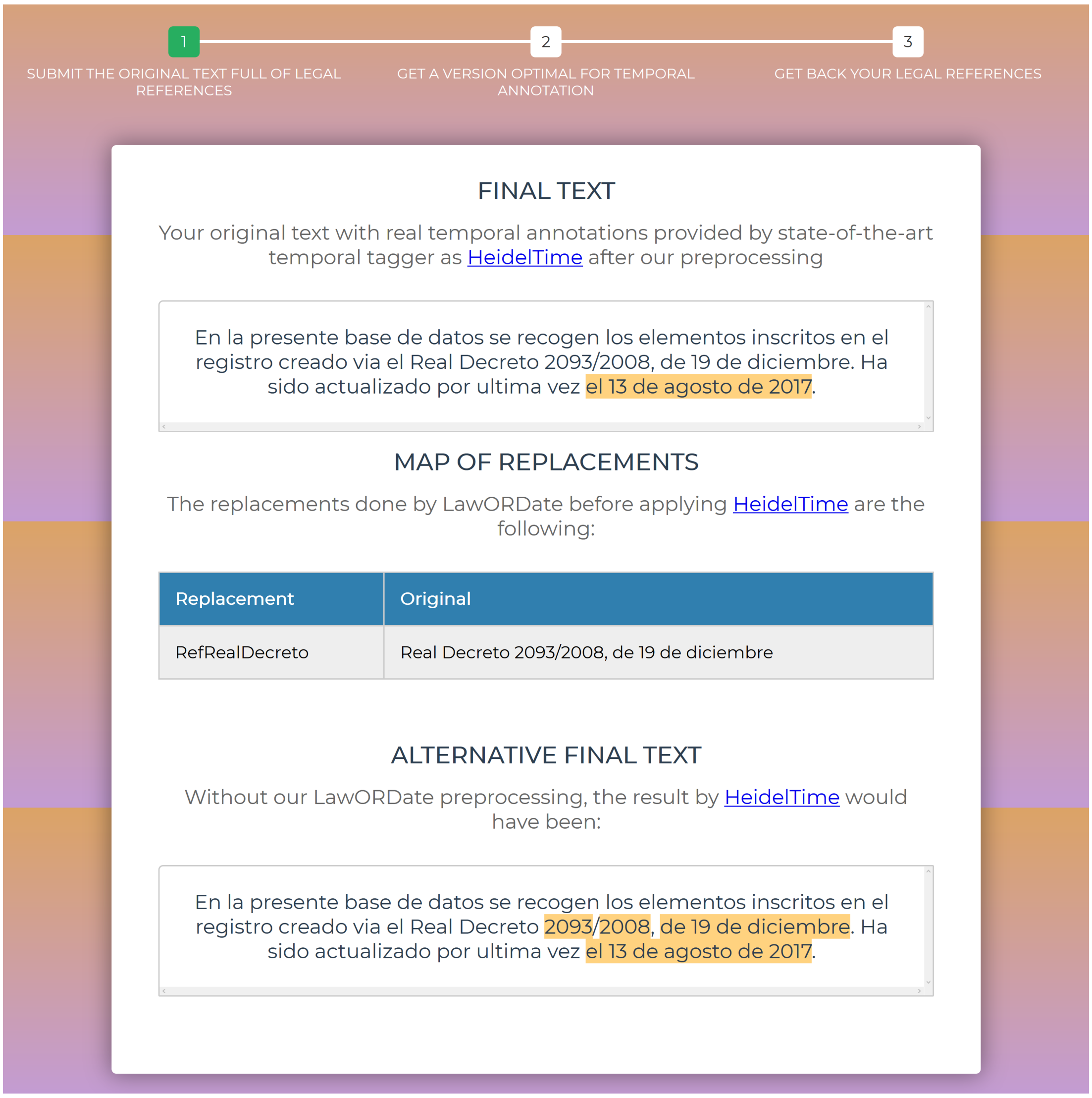


Figure 6.6: Screenshot of the tool lawORdate. The first text area includes the annotation done by the state-of-the-art temporal tagger HeidelTime (Strötgen and Gertz (2013)) after applying the replacements. Then the map of the replacements is done, and finally the result of the tagger without using lawORdate replacements. The translation of the example text is: "Database that provides information on Technology Centers and Innovation Support Centers registered in the registry created by the Royal Decree 2093/ 2008, of December 19. Last update on **August 13, 2017**.". While lawORdate just marks the temporal expression in bold, not using it would have produced the tagging of the expressions underlined (that are part of a legal reference, and not all of them even temporal expressions).

actually happened in November according to our calendar. Additionally, some countries have their own calendars and date elements, such as the Japanese era system, the Chinese lunisolar calendar or the Persian Solar Hijri calendar.

TE dependent on geographical information Geographical information can refer to physical geography or to political geography information. An example of the former is *spring*, which depends on the hemisphere, and an example of the latter is *Día del Niño* (*International Children's Day*), which depends on a political decision different for every jurisdiction –it is for instance celebrated the 15th of April in Spain, but the 30th in Mexico. Geographical information may also help to correctly normalize certain date formats, since 09/10/2019 means 9th October in Europe but 10th September in the United States. Finally, dialects also imply different ways to refer to time, such as the Latin American expression "cinco para la una" ("five to one"), that is not used in Spain (where it is usually expressed as "la una menos cinco", "one minus five").

TE dependent on the register The jargon can also affect TE identification and normalization. There are a lot of expressions in Spanish where non-temporal words are used in a temporal sense, some of them included in the HourGlass corpus. Examples of these are the expressions "Él tiene 37 castañas" ("He has 37 chestnuts") and "Él tiene 37 tacos" ("He has 37 tacos"), both meaning "He is 37 years old". Other expressions can also change their meaning in a meronymic way, such as is the case of "Tiene 30 primaveras/abriles ya" ("He already has 30 springs/Aprils"), where a part of the year (the spring or the month of April) represents the whole year. Similarly, we also have many idioms involving temporal expressions that should not be tagged, such as the phrases "Hasta el 40 de mayo no te quites el sayo" ("Until 40th May, do not take off the jacket", meaning that the beginning of June can still be chilly), "En abril, aguas mil" ("In April, a thousand waters", meaning that April is usually rainy) or "A buenas horas mangas verdes" ("At good hours, green sleeves", meaning someone acted too late), and expressions like "en el último minuto" ("in the last minute", meaning close to a deadline), where "minute" should not either be tagged. Regarding Latin American Spanish, there also exist a lot of similar idioms and expressions, such as "la hora del moro" ("the hour of the moor"), which means "lunchtime" in the Dominican Republic.

Despite the problem of resolving CDTEs has already been partially studied (Lee et al., 2014), to the best of the doctoral candidate knowledge there are no full-working solutions.

Summary

This last chapter of Part II, related to temporal expressions, presented Añotador, a temporal tagger for Spanish and English that (1) covers untackled particularities of the Spanish language, including Latin-American ones, and (2) has a special implementation for the legal domain. Also some real use cases where Añotador was successfully used were reviewed (Section 6.1.5), and evaluated it against different corpora (Section 6.1.6). This software has also been registered in the Registro Territorial de la Propiedad Intelectual de la Comunidad de Madrid[136], the intellectual property office from the region of Madrid, under a GPL-3.0 license.

Additionally, a tool that allows the user to preprocess citations that can be misleading to temporal taggers (lawORdate, Section 6.2) was introduced, and an analysis of open lines of research in the temporal tagging domain beyond this thesis scope are outlined (Section 6.3).

The performance of Añotador in legal texts, together with the analysis presented in Chapter 3 and the tool lawORdate, confirm hypothesis H1.a and H1.b.

[136] https://www.comunidad.madrid/gobierno/informacion-juridica-legislacion/registro-territorial-propiedad-intelectual

Part III

EVENTS

Chapter 7

Corpus of Events

Similarly to what happened when dealing with temporal expressions, corpora annotated with events in the legal domain is scarce. There are no available corpora that can be used for event extraction in this domain, and hence they had to be created for the purpose of this thesis. In addition to this scarcity, it must also be noted that the definition of event is not as consensual as that of temporal expression, which hinders substantially the task of deciding what is an event and what is not, tending to devolve this issue to the requirements of a target task or to a specific domain. Therefore, we find among available corpora more variety of annotation standards and notions of events.

Since for the temporal expression extraction the TimeML standard was adopted, the logical step would be to follow the same schema for the task of event extraction. Nevertheless, a first test consisting of running several state-of-the-art temporal taggers on several court decisions ruled out this option. The reasons why TimeML was discarded, together with the results of the test, are explained below.

The first reason is the abundance of events tagged. Almost any verb is considered an event, and therefore the amount of annotations is high, as can be appreciated in Table 7.1a and Table 7.1b. The first table contains the amount of events of each type (regarding TimeML standard) tagged by four state-of-the-art temporal taggers on the TempCourt corpus (presented in Section 5.1), while the second indicates the POS tagging information of the words included in the event annotations. Just regarding the *all* row of each temporal tagger, which indicates the amount of events tagged, it becomes evident that there is a lot of variability on the consideration of what is an event and what is not, even following the same guidelines. This is even more obvious when checking the tables in Figure 7.1, which show the agreement among the different taggers in the TempCourt corpus, that is not high, especially in the USSC corpus.

Moreover, comparing the TempCourt numbers (the amount of tokens and average tokens per sentence were indicated in Table 5.1) to the number of events per document, it is derived that on average, there is almost one event per sentence, and that between the 7 and the 9% of the tokens in a document (12 and 13% in the case of TARSQI) are considered events, what is an extremely high ratio. In fact, the calculus per sentence results between 0,99 and 1,59 events per sentence for the ECHR part of the corpus, between 2,26 and 4 events per sentence for ECJ and 1,43 and 2,31 events per sentence for USSC, which shows that on average every sentence has at least one event, and even more in the case of documents with longer sentences.

ECHR		CAEVO	CLEARTK	TARSQI	TIPSEM
	CAEVO		0,76	0,56	0,72
	CLEARTK	0,76		0,58	0,79
	TARSQI	0,56	0,58		0,55
	TIPSEM	0,72	0,79	0,55	

ECJ		CAEVO	CLEARTK	TARSQI	TIPSEM
	CAEVO		0,69	0,48	0,69
	CLEARTK	0,69		0,50	0,73
	TARSQI	0,48	0,50		0,52
	TIPSEM	0,69	0,73	0,52	

USSC		CAEVO	CLEARTK	TARSQI
	CAEVO		0,25	0,52
	CLEARTK	0,25		0,17
	TARSQI	0,52	0,17	

Figure 7.1: Agreement among temporal taggers in the event annotation task. The calculus has been done dividing the matched expressions by the sum of the total annotations of each pair of taggers. The color of each cell is a gradient between zero (red) and 1 (green), with yellow in the middle.

The second problem found is the incorrect annotations of legal domain events. The second paragraph of the introduction of Case C249/13 of the European Court of Justice (that is part of the TempCourt corpus, document 62013CJ0249.xml) will be taken for

		ECJ*		ECHR		USSC	
		total	average/doc	total	average/doc	total	average/doc
CAEVO	occurence	1933.0	322.17	475.0	47.5	3322.0	332.2
	perception	7.0	1.17	8.0	0.8	13.0	1.3
	i_action	112.0	18.67	20.0	2.0	287.0	28.7
	reporting	32.0	5.33	12.0	1.2	83.0	8.3
	aspectual	2.0	0.33	1.0	0.1	8.0	0.8
	i_state	99.0	16.5	33.0	3.3	226.0	22.6
	state	63.0	10.5	5.0	0.5	123.0	12.3
	all	2248.0	374.67	554.0	55.4	4062.0	406.2
TARSQI	occurence	4778.0	597.25	798.0	79.8	5721.0	572.1
	perception	99.0	12.375	11.0	1.1	106.0	10.6
	i_action	97.0	12.125	21.0	2.1	223.0	22.3
	reporting	56.0	7.0	10.0	1.0	144.0	14.4
	aspectual	6.0	0.75	2.0	0.2	33.0	3.3
	i_state	67.0	8.375	22.0	2.2	126.0	12.6
	state	209.0	26.125	24.0	2.4	196.0	19.6
	all	5312.0	664.0	888.0	88.8	6549.0	654.9
CLEARTK	occurence	1890.0	315.0	437.0	43.7	3227.0	322.7
	perception	36.0	6.0	10.0	1.0	106.0	10.6
	i_action	157.0	26.17	52.0	5.2	524.0	52.4
	reporting	56.0	9.33	28.0	2.8	136.0	13.6
	aspectual	12.0	2.0	5.0	0.5	26.0	2.6
	i_state	121.0	20.17	24.0	2.4	210.0	21.0
	state	61.0	10.17	5.0	0.5	110.0	11.0
	all	2333.0	388.83	561.0	56.1	4339.0	433.9
TIPSEM	occurence	1742.0	290.33	406.0	40.6		
	perception	63.0	10.5	4.0	0.4		
	i_action	224.0	37.33	67.0	6.7		
	reporting	52.0	8.67	26.0	2.6		
	aspectual	19.0	3.17	3.0	0.3		
	i_state	143.0	23.83	33.0	3.3		
	state	73.0	12.17	3.0	0.3		
	all	2316.0	386.0	542.0	54.2		

Table 7.1a: Event annotations in the TempCourt corpus by the state-of-the-art temporal taggers CAEVO, TARSQI, ClearTK and TIPSEM. Per each part of the corpus, the total amount of each type of events is given, as well as the overall *all*. (*) The ECJ part of the corpus includes just six documents because some taggers had troubles with the rest of them.

		ECJ*		ECHR		USSC	
		total	percentage	total	percentage	total	percentage
CAEVO	adj	92.0	4.09%	7.0	1.26%	105.0	2.59%
	noun	343.0	15.26%	33.0	5.96%	400.0	9.85%
	verb	1807.0	80.38%	512.0	92.42%	3501.0	86.19%
	other	6.0	0.27%	2.0	0.36%	44.0	1.08%
TARSQI	adj	232.0	4.37%	28.0	3.15%	252.0	3.85%
	noun	2200.0	41.42%	280.0	31.53%	2179.0	33.27%
	verb	2880.0	54.22%	579.0	65.20%	4114.0	62.82%
	other	0.0	0.0%	1.0	0.11%	29.0	0.44%
CLEARTK	adj	60.0	2.57%	6.0	1.07%	81.0	1.87%
	noun	389.0	16.67%	28.0	4.99%	460.0	10.60%
	verb	1876.0	80.41%	520.0	92.69%	3719.0	85.71%
	other	8.0	0.34%	7.0	1.25%	87.0	2.01%
TIPSEM	adj	47.0	2.03%	8.0	1.48%		
	noun	420.0	18.14%	48.0	8.86%		
	verb	1849.0	79.84%	486.0	89.67%		
	other	0.0	0.0	0.0	0.0		

Table 7.1b: POS tagging category of the words annotated as events in the TempCourt corpus by the state-of-the-art temporal taggers CAEVO, TARSQI, ClearTK and TIPSEM. Per each part of the corpus, the POS tagging (*adjective, noun, verb* or *other*) of the words annotated as events, as well as the percentage each category represents. (*) The ECJ part of the corpus includes only six documents because some taggers had troubles with the rest of them.

illustration.

> *"REQUEST for a preliminary ruling under Article 267 TFEU from the Tribunal administratif de Pau (France), made by decision of 30 April 2013 , received at the Court on 6 May 2013, in the proceedings"*

In this excerpt there is a mention to *preliminary ruling*, that is a specific legal procedure in the European Union, but just *ruling* is tagged by the temporal taggers. In fact, since the actual event happening is the *request for a preliminary ruling*, *request* should be tagged as a part of the same event. A special case of these events including several verbs are the ones including light verbs, such as *make a decision*, which have a really high presence in legal texts, and should be taken into account when extracting events from them and be considered as a single event. An example of this are the two following constructions, which mean exactly the same, and should therefore be tagged equally: *make a request for a preliminary ruling* and *request for a preliminary ruling*. Similarly, later in the text there are mentions to a *hearing*, that is a relevant event in the legal domain, but it is not always tagged by temporal taggers (in fact, just TIPSEM tags it so).

Finally, the TimeML standard covers information related only to time, so besides the type of event and a possible link to other temporal annotations, temporal taggers following TimeML just provide grammar information and considerations like *polarity* or *tense*, but not contextual information such as who did the action, or any relevant surrounding circumstances to the tagged event, such as modifiers, complements or objects attached to it. The TimeML approach to events can be therefore useful for general short texts, such as news, but not for the legal domain, with long texts including a lot of verbs on the one hand, and full of unconsidered domain events on the other.

These are the reasons why the TimeML standard was not used in relation to event annotation of corpora. The annotation schema used instead, presented in the following sections, was kept simple and generic in order to be easily used or transformed to other possible representations, but covers the most relevant information with regard to events according to the legal experts that annotated the texts.

7.1 EventsMatter

The EventsMatter corpus comprehends thirty documents from the European Court of Human Rights, where the most relevant events and their contexts were annotated. This corpus is the result of a collaboration with legal researchers Erwin Filtz and Cristiana Santos, and was supervised by Sabrina Kirrane and Axel Polleres. The corpus and the different approaches tested on it are available online[137] under a GNU General Public License 3.0, and explained in a paper published in the proceedings of the 33rd International Conference on Legal Knowledge and Information Systems (JURIX 2020), one of the most important conferences in the intersection of Law, Artificial Intelligence and Information Systems (Filtz et al., 2020). Part of the information, data and images from that publication are reused in this section, as well as from the guidelines of the corpus[138].

Similarly to the TempCourt corpus (Section 5.1), the corpus EventsMatter was created to cover a gap in the legal domain, since there is no public corpus available annotated with events, relevant or not. During its creation, additionally, an analysis of the minimum annotation schema needed to cover relevant events, as well as a guideline to help the process of annotation, were performed and written as a starting point of future research in the domain. This led eventually to the creation of an ontology that will be presented in Section 9.1.

The remainder of the chapter is as follows. Section 7.1.1 introduces how the data was collected. Section 7.1.2 explains how the manual annotations were performed. Section 7.1.3 shows some examples of the difficulties found during the annotation process. Section 7.1.4 details the final format of the document and how it has been made available to the research community. Finally, Section 7.1.5 shows some statistics on the corpus.

7.1.1 Data collection

On the basis of the experience with the TempCourt corpus, the first steps in building EventsMatter were to define the source of the documents and the annotation schema.

Regarding the source, in order to justify the choice, first

[137] `https://mnavasloro.github.io/EventsMatter/`

[138] `https://mnavasloro.github.io/EventsMatter/Guidelines.pdf`

For this reason, it was decided to restrict the documents in the corpus to a unique court, namely the European Court of Human Rights (previously described in Section 5.1.1). This court was chosen because its documents contain:

- Different types of time-related events concerning different actors in comparison with the decisions of the Court of Justice of the EU (Navas-Loro and Santos, 2018).

- A standard structure in which different legal events are embedded. ECHR decisions are divided into several sections containing specific information according to Rule 74 of the Rules of the Court (Registry of the Court, 2020):

  i) Preamble

  ii) The facts, with the identification of the parties;

    ii.i) Refers to a summary of the submissions of the parties comprising their main legal arguments;

    ii.ii) Relevant domestic law. It encompasses provisions of domestic law, and/or other pertinent international or European treaties.

  iii) Complaints

    iii.i) The Law. It comprises the merit of the case, and, i.e., meaning the reasons in "point of law" articulated by the Court and operative provisions thereof. Herein are stated the alleged violation(s) of the article(s) of the Convention.

    iii.ii) Remaining Complaints

  iv) Decision

Also to focus on a very specific topic when retrieving the documents, as recommended by external experts in the legal domain, was decided. Due to the interest in the domain and in the Lynx project, and also the expertise of the law expert collaborating in the work (Cristiana Santos), we decided on the privacy topic, and namely looked for court decisions citing Article 8 of the Convention (Right to respect for private and family life). Nevertheless, as will be seen later, this does not limit the appearance of many different verbs that at first sight would look unconnected to the choice. Concerning the size of the corpus, we collected the same amount of documents as in TempCourt, thirty documents. The statistics of EventsMatter in terms of the number of tokens, document size and sentence length are detailed in Section 7.1.5.

On the other hand, with regard to annotation we decided to annotate in the first round the extent of the core of the event (mainly the event) and when it happened, and then, in subsequent annotations, add or extend the annotations based on the discussion among annotators.

7.1.2 Annotation Methodology

The EventsMatter corpus was annotated by two legal experts in several iterations using the software GATE (Cunningham et al., 2013). The experts annotated independently and then met with a third person to reach a consensus on the disagreements. Together with this consensus annotation set, the original annotations of both annotators are available in the corpus so that differences can be consulted.

Regarding the concept of relevance taken into account and the guidelines given to the annotators, we decided to focus on event extraction aimed at automated court decision timeline generation. We were therefore interested in information that is relevant to searching for or extracting time-related information, such as events, processes, temporal expressions, and the parties involved. As time-related events of cases are linguistically expressed, we annotated the most salient candidate passages thereof. The following section explains the annotation methodology and the specific guidelines supporting the annotation task.

7.1.2.1 Annotation

Judgments were manually annotated following the frame "who-when-core event". To illustrate the applicability thereof, an annotated paragraph of the case Altay v. Turkey (no. 2), no. 11236/09, 9 April 2019 (a case on the respect of private life) will be used:

> *"On 29 May 2008 the applicant lodged an application with the Edirne Enforcement Court for the restriction on the conversations between him and his lawyer to be lifted."*

Who corresponds to the subject of the event, which can either be a subject, but also an object (i.e., an application); in the example, the subject is "(the) applicant";

When refers to the date of the event, or to any temporal reference thereto, that can be a reference of another event (e.g., *After the death of the widow, X happened*). In the paragraph considered, the "when" is the "29 May 2008";

What usually corresponds to the main verb reflecting the baseline of all the paragraph (which in this case is "lodged"); additionally, we include thereto a *complementing* verb or object whenever the core verb is not self-explicit or requires an extension to attain a sufficient meaning of the core event; in the paragraph considered, the core event is "lodged an application". Another example would be "dismiss an action".

Event relates to the extent of text containing contextual event-related information. The *type* of such annotations can be either *circumstance* – meaning that the event corresponds to the facts under judgment; or *procedure*– wherein the event belongs to the procedural dimension of the case. This includes court procedures (legal proceedings stricto sensu), but also actions that trigger procedural effects. This distinction is derived from the analysis of particularities described in Chapter 3. In the paragraph at stake, we annotated the sentence as an Event of the type *procedure*.

7.1.2.2 Guidelines

In addition to the event components described in the former section, we have annotated related-time events with concrete guidelines shown below:

Extension of core event. One core event can also include two or more close-related verbs, e.g. "divorced" and "agree on custody", instead of annotating two connected verbs autonomously. Moreover, whenever there is a causal relationship between events, we annotate merely one, e.g. "they drink water and they felt unwell".

Repeated events. When there is a reference to an event happening on several dates (e.g. the dates of the birthday of three applicants, respectively), we annotate solely one event as the core, and count with one annotation that covers all the related dates.

Non-dated events. Events that are not dated, though semantically expressing an implicit time reference, are then annotated under "when". Examples of this would be time expressions as "the same date", "this afternoon", "on unspecified dates", "number of occasions".

Non-annotated events. Some events were not considered relevant to be depicted in a timeline, and therefore not annotated, e.g. the fact that *X was born in Y* is usually irrelevant.

Factuality. Events that are named but do not occur, are yet annotated, but they are marked under "factuality" feature to be distinguished, but not included in the timeline. When events are negated, this feature equals to "NOT"; for instance, when it is mentioned that a party does not appeal against a decision.

Importance. In some cases, the annotators did not agree on adding or deleting an event from the annotation in the consensus round. When this happened, the event (considered relevant by just one of the annotators) was marked with the feature *importance:L* (from *low*).

Furthermore, other criteria were taken into account during the annotation process and are illustrated below:

- The first and last events in a case are always annotated the same:
  - First event (exposing the applicant and the information of the case): always follows the structure "The case originated in an/**N application(s)** (no. X) against X **lodged** (...) by (...), on **DD MM YYYY**". There can be one or several applicants, and therefore one or several dates. In all the cases, the dates will be the *when*, lodged will be the *what* and *application* or *# applications* will be the *who*.
  - Last event (exposing the final decision): the *who* is THE COURT, the *what* are the different decisions (one *what* annotation each, separately), the event goes from "FOR THESE REASONS, THE COURT" to the last sentence before the signatures.

- Not to annotate irrelevant determiners (*the, a*) at the beginning of the *who* event component.

- When we have a mention to an event that happens several times: ONE annotation for the core and ONE annotation that covers all the dates.

- The *what* annotation can be discontinuous in an event, and can encompass several verbs if they are the result of the same action (e.g., *The Court* ***communicated the decision*** *in September 2010,* ***upholding the previous judgment***..

- Events that do not happen are annotated but they are marked with *factuality* feature to be distinguished and not included in the timeline.

Some examples of difficult annotations that inspired some of the guidelines are shown and analyzed in the next section.

7.1.3 Main difficulties found

During the annotation process, some events were difficult to tag, while others sparked discussion about how to do so, challenging the stipulated guidelines and evidencing how complex and subjective annotating tasks can be. Herewith some annotations that triggered discussions on the type of events (*procedure/circumstance*) are shown. The respective disputable cases are transcribed and then commented on the achieved consensus.

Procedural types of related events.

- "On 1 August 2000 the Ministry of the Interior of Belarus ordered the applicant's arrest on suspicion of her having committed several criminal offences".

- "On 28 March 2014 the Town Court extended the applicant's detention until 18 August 2014".

- "On 20 August and 21 September 2015 the investigative committee of the Republic of Belarus discontinued the criminal proceedings against the applicant".

- “The applicant complained before the domestic courts about the lawfulness of the interception of his phone conversation and the accuracy of the transcript. However, the High Court merely replied that the impugned interception had been lawful and within the scope of Law no. 51/1991.”

In the cases shown, the events do not refer to a court procedure per se, though it triggers legal procedural effects.

Circumstantial types of related events:

- “In August and September 2010 the applicant lodged two complaints with the Press Complaints Commission (Pressens Faglige Utvalg) against two publications owned and controlled by Mr Trygve Hegnar: the weekly and daily business newspapers Kapital and Finansavisen”;
- “On 20 February 2003 the applicant, a bank manager at that time, was placed in pre-trial detention by the Bucharest Anti-Corruption Department of the Prosecutor’s Office, on a charge of taking a bribe in order to favourably influence the acceptance of a loan requested by M.G.”;
- “On 17 November 2014 the Sunzhenskiy District Bailiffs’ Service in the Republic of Ingushetiya refused to institute enforcement proceedings since the debtor, R.M., resided in Moscow”.

In the cases shown above, the event corresponds to the facts under judgment (*circumstance* events). In these statements, the annotators are not aware of the actual jurisdictional competence of the local system relative to the case (“Press Complaints Commission”, the “Bucharest Anti-Corruption Department of the Prosecutor’s Office”, or the “Sunzhenskiy District Bailiffs’ Service”) to qualify as procedural type event.

- “On 26 February 2014 the Deputy Town Prosecutor carried out an inspection of remand prison SIZO-6”.

The issue relates to the semantics attributed to the role “Deputy Town Prosecutor” which renders the idea of being a court magistrate, and as such, it would be deemed as a procedural event. Herein, the function instead refers to an inspection task, without procedural effect.

- “The applicant claims, and this has not been disputed by the Government, that the media to be returned to the applicant contained, inter alia, legal advice protected by lawyer-client privilege”.

The verb used (“claims”) means “argues”, not referring to a legal procedural action.

- “According to publicly available information, after the death of a baby born at home in June 2011, the police started a criminal investigation”.

The paragraph refers to the starting of a criminal action, not of a procedural court event.

7.1.4 Format

The final documents have been generated in several formats[139]. First, as GATE XML documents, that facilitate the storage of different annotation sets (for the different annotators and the final *consensus* set) and also the visual and numerical comparison of the different sets. Second, a set of clean .xml documents with the raw text is provided in order to facilitate annotation by other systems. Finally, all original .docx documents of the decisions are also given in order to use its inner structure for processing the different parts of the judgment.

	Tokens	**Avg. Tokens per Doc**
What	721	24,03
When	528	17,6
Who	418	13,93
Procedure	294	9,8
Circumstance	320	10,67
Events	615	20,5
Total	66970	2232,33

Table 7.2: EventsMatter corpus statistics.

[139]The final corpus can found at Zenodo: `https://zenodo.org/record/4032617`

7.1.5 Statistics on the Corpus

Table 7.2 presents the statistics of the corpus, where we can see that each document has an average of around 20 events. The *What* annotations are usually longer than the others, and *Who* is not always present in the sentence. Regarding the share of types of events, there are slightly more *Circumstance* events than *Procedural* ones, but it's very balanced.

Summary

In this chapter EventsMatter, the first corpus of legal documents annotated with relevant events was presented. After introducing some statistics on events within judgments, the different steps in the creation of the corpus were described. First, the data collection; then, how the corpus was annotated and the difficulties found. Finally, the format and the statistic of the corpus were discussed. The annotation of the corpus by different annotators following the proposed guidelines confirmed the hypothesis H2.a of the thesis.

Next chapters will present the remaining work done with regard to events: the event extraction tools created (Chapter 8) and the event-related resources (Chapter 9).

Chapter 8

Event Extraction

Ludwig Wittgenstein opened his *Tractatus Logico-Philosophicus* observing that the world is the totality of facts, not of things –quite a reasonable observation for a logician who was interested in the truth of propositions at that time. When evaluating the events described in a legal decision, focusing on the events and their logical sequence seems also very reasonable and the storyline is of pivotal importance. In this chapter, it is assumed that a judgment can be described as a series of time-marked happenings (*events*) instead of focusing on the other entities (things).

Prior to the development of software able to perform this transformation from judgments to events, during the research stay done at the National Institute of Informatics in Japan (NII), a first event extraction framework was developed in order to tackle a very specific problem in the legal domain. This system, named ContractFrames, can be considered the first approach to the task, so it will be briefly presented in Section 8.1. Afterwards, the main result of the thesis regarding event extraction, called WhenTheFact, will be presented in more detail in Section 8.2.

8.1 ContractFrames

ContractFrames is the result of the research stay at the NII, and the information of this section is partially based on the derived publication (Navas-Loro et al., 2019b).

8.1.1 Context Problem

PROLEG (Satoh et al., 2011) is a legal reasoning system that formalizes legal information in the form of logical predicates, being afterwards able to reason with it. This system is able to represent and reason about the status of a contract and derive information such as its validity or the right or reason of a rescission, but not to extract the information from natural language. In order to automatically transform the input text into logic facts, a bridge between NLP and this logical system for automatic retrieval of all relevant facts from text to populate the PROLEG fact knowledge base has been developed under the name of *ContractFrames*.

This framework is able to translate natural language texts referring to the different statuses of a purchase contract into PROLEG clauses. These texts are not normative texts nor regular texts (being both types extensively studied in previous literature), but some natural language text at a midpoint between regular language and pure legal language; an example of one of these texts can be found in Fig. 8.1, along with its translation into PROLEG.

'person A' bought this_real_estate from 'person B' at the price of 200000 dollars by contract0 on 1/January/2018. But 'person A' rescinded contract0 because 'person A' is a minor on 1/March/2018. However, this rescission was made because 'person B' threatened 'person A' on 1/February/2018. It is because 'person B' would like to sell this_real_estate to 'person C' in the higher price. So, 'person A' rescinded rescission of contract0 on 1/April/2018.

```
minor(personA).
agreement_of_purchase_contract(personA,personB,this_real_estate,200000,2018 year 01 month 01 day,
    contract0).
manifestation_fact(rescission(contract0),personA,personB,2018 year 03 month 01 day).
fact_of_duress(personB,personA,rescission(contract0),2018 year 02 month 01 day).
```

Figure 8.1: Example of an input text of ContractFrames and its correct output, in the format expected by PROLEG. Example extracted from Navas-Loro et al. (2019b).

8.1.2 Approach Used

In this subsection, the approach used for the task will be introduced.

8.1.2.1 Frame definition

To target the translation between the natural language text and the expected clause format, a frame-based approach[140] was chosen. To this aim, I analyzed the clauses, derived the situations they represented and the relevant information related to each of them, and then created the frames, depicted in Fig. 8.2.

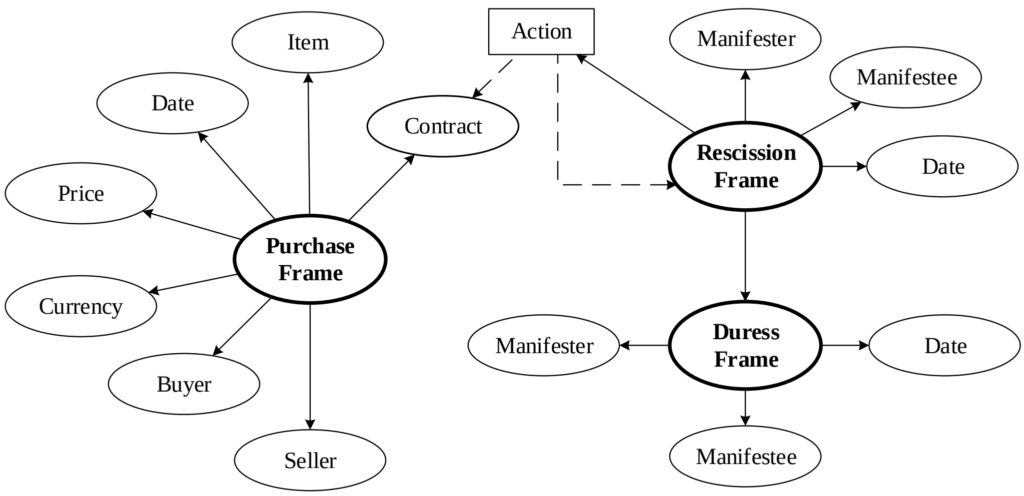


Figure 8.2: The three frames handled in ContractFrames (Purchase, Rescission and Duress) and how they interact. An action can be a *contract* or a *rescission*, therefore a *rescission* can be of a *contract* or of another *rescission*. A *duress* is also necessarily attached to a *rescission*. These frames were extracted from the PROLEG predicates, so they are sufficient for modeling information involved in the task and efficient for this specific work, but not possible to generalize (Image from Navas-Loro et al. (2019b)).

8.1.2.2 Challenges found

The main issues found during the processing of the text are summarized below (they are explained in more detail in Navas-Loro et al. (2019b)).

[140]According to Marvin Minsky (Minsky, 1975), a frame is *'a data-structure for representing a stereotyped situation'*.

- *Style of the text*, using expressions that might mislead NLP tools, such as in the example *"A sells L to B by Part C"*, where the letters *A, L, B* or *Part C* should be considered Named Entities. Preprocessing the text helped dealing with these situations.

- *Relevance*, since not all the sentences contain important information, and not in the same degree.

- *Factuality*, in some cases the text exposes not only facts, but also possibilities, preferences or intentions. These distinctions should be clearly made.

- *Paraphrasing and Complexity*, since there are many ways to express the same information and no resources in the legal domain to deal with this. Additionally, as previously reported in literature (Dell'Orletta et al., 2012), legal texts tend to be more complex than those from other domains (having higher parse trees, more words per sentence and different POS distribution, among others).

- *Coreferences and nesting*, not all the information is necessarily contained in a single sentence, and different manifestations of the same frame can appear in the same text, increasing the correct resolutions of coreferences. Also, in some cases, the argument of some of the frames must be derived from related or nested frames, since it is not explicitly mentioned in the text.

- *Matching*, since some frames are dependent (a *duress* frame must be related to a *rescission* frame), they must be correctly interpreted and matched.

The detection strategy, presented in the following section, was developed taking the aforementioned obstacles into account.

8.1.2.3 Detection strategy

The framework *ContractFrames* makes use of the NLP tool Stanford CoreNLP (Manning et al., 2014) with TokensRegex (Chang and Manning, 2014), that allows us to build rules based on semantic considerations. The steps of the algorithm are explained below. A more detailed explanation that also refers how the issues from the previous section were solved on each step is available in the main publication of this result (Navas-Loro et al., 2019b).

1. Preprocessing the input text: the first step in the framework deals with problems related to the style of the text and the different information. The objective of this step is to output a version of the text easier to be understood by the CoreNLP pipeline. Some of the actions taken are algorithms that replace and handle misleading expressions (e.g. A, or *'person B'*), standardize the dates (since the format '`dd/Month/yyyy`' is not recognized by CoreNLP) or detect clauses with arity one such as `minor(agent)`.

2. Annotation with the CoreNLP pipeline, including the rules to detect different kinds of events (establishment of contracts, purchases, sales, rescissions, duress...) both in verbal forms (*buy, sell, rescind*) or as noun events (*purchase, sale, rescission*).

3. Parse sentence by sentence, and token by token.

   (a) If the token is an event, it is checked if it is a fact (not negated, nor an intention or a possibility). Once verified, which type of frame it is is checked and then different rules are applied to the dependency parsing of the sentence to find each of its arguments (if available).

   (b) If the token has any other relevant annotation by the rules, it is stored as relevant information in the sentence.

   Once each token has been analyzed, any missing information in frames that have been initialized due to some found event is completed.

4. Once the whole sentence has been processed, the information stored in the frames is checked. If there is still any information missing, it will be looked for explicitly, and also if new information can complete previous frames (nesting problem)is checked.

5. Finally, once all the sentences have been processed, the information on the final frames are completed using some common sense.

6. Last step involves transforming the information in the frames in PROLEG clauses, including reversing the replacement done during the preprocessing step.

Although the CoreNLP coreference tool was initially included in the system, it presented some issues when dealing with the same event referred both by using nouns and verbs[141]. For this reason, it was decided to develop an algorithm to detect previous potentially similar events and merge them, which is executed in step 4. Fig. 8.3 depicts the pipeline of ContractFrames.

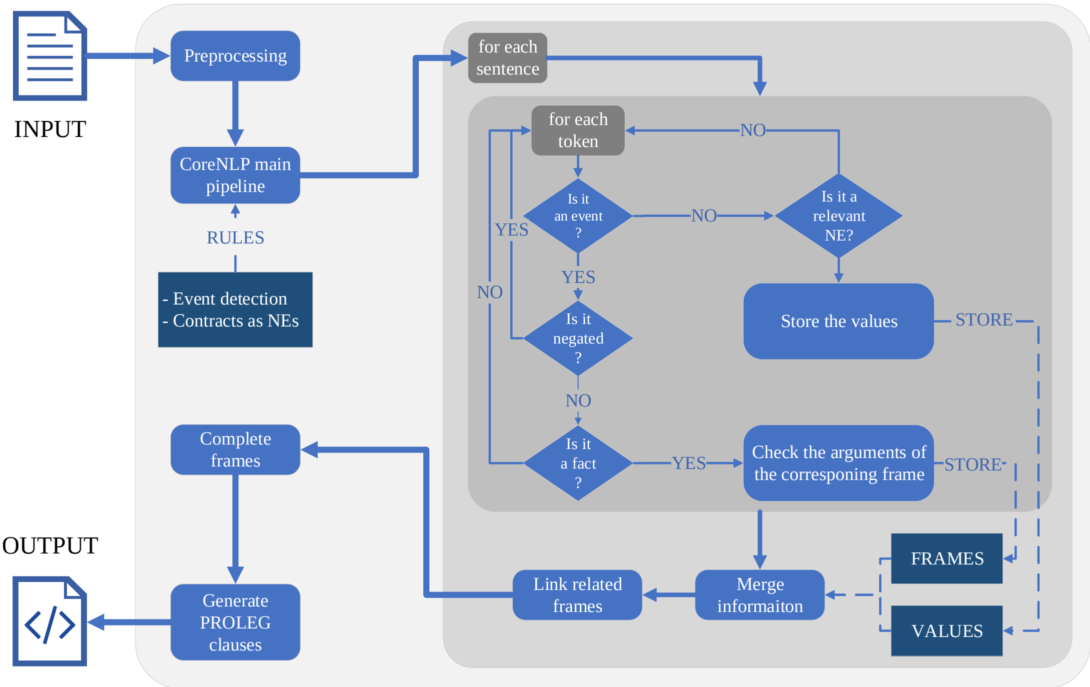


Figure 8.3: Pipeline of ContractFrames.

8.1.2.4 Availability

Regarding the test, since there were no corpora available to test the framework, legal experts were asked (not involved in the development of the framework) for texts in the format that the systems should be able to parse. The texts provided include both semantic and syntactic paraphrasing, as well as different levels of nesting and events are represented in this dataset, that includes texts of different length explaining the workflow of a contract, and even surrounding facts not exploited by the system. This

[141] While the tool could detect that for instance that in *"The rescission of the contract was done on 1 February, 2018. This rescission was cancelled later"* there was a coreference, it did not succeed in cases such as *"A rescinded the contract with B. This rescission was cancelled later"*.

dataset is provided with the code of ContractFrames[142], that is publicly available in a GitHub repository[143].

Besides the logic clauses output format, the system also generates an XML output that allows the visualization of the inner custom annotations in the text, namely events and named entities like contracts. An example of this visualization (using the tool GATE (Cunningham et al., 2013)) can be found in Fig. 8.4. Additionally, an ontology that expresses contract statuses has been developed. This ontology, called Contract Workflow Ontology[144], is capable of representing the different types of events processed, such as agreements and rescission, as well as others in the workflow of a general event such as negotiation. A method for generating an output in the form of triples is provided in the system.

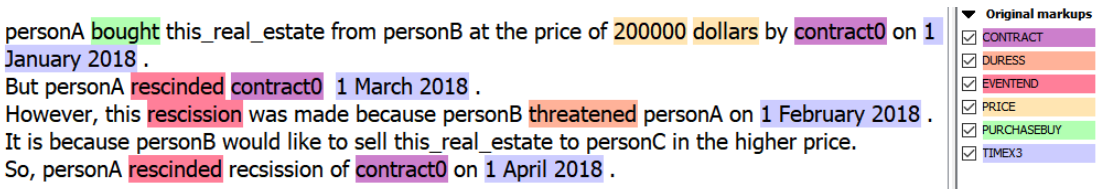


Figure 8.4: Visualization of custom annotations in XML.

Since this line of research focused on a very specific and limited problem, that at the same time required external help in order to provide examples to analyze, it was decided to focus on more generic and available types of documents, namely judgments, and to go for a more general approach that could be easily adapted to new corpora. The result of this task is presented in the next section.

8.2 WhenTheFact

Before undertaking the event extraction task, an analysis of the previous approach in the legal domain was carried out (Navas-Loro and Santos, 2018). One of the suggestions made during the presentation of this work was to take into account the discourse extraction when dealing with events relevance. This has been taken into account, and

[142] https://mnavasloro.github.io/ContractFrames/
[143] https://github.com/mnavasloro/ContractFrames
[144] https://mnavasloro.github.io/ContractFrames/datamodel.html

it is further discussed in Section 8.2.1, while the training strategies of the system built are explained in Section 8.2.2 and the extraction is described in Section 8.2.3.

Additionally, the differences detected among courts in previous corpora annotation works led to the choice of one of them for implementation. This choice also invited to a first discourse analysis module dependant on the kind of document, that selects the relevant parts of the texts that the event extractor core will work on. In order to show that the event extraction method presented is easily generalizable, two different sources to retrieve legal documents were used, namely the European Court of Human Rights (ECHR) and the European Court of Justice. Also, the approach was not limited to the detection of the trigger word (or core) of the event, but also when it happened and who did it, if available, following the EventsMatter corpus format. A very basic version of this work was briefly introduced in a conference paper (Filtz et al., 2020).

8.2.1 Structure Extraction

To illustrate the importance of structure extraction when dealing with relevant events, let us analyze their presence along the different sections of the documents in the Events-Matter corpus (Section 7.1). Figures in this section represent the distribution of events along paragraphs and sections (Fig. 8.5), in the paragraphs of an specific section (Fig. 8.6), per section (Fig. 8.7a) and on average per section (Fig. 8.7b).

Fig. 8.5 depicts the distribution of events along each of the thirty documents in the EventsMatter corpus. Regarding the colours, since not all the judgments have the same amount of paragraphs per section, white means there is no such paragraph in that document. The lightest blue indicates the paragraph exists, but contains no events, while darker blue until purple denotes the existence of one or more events (until six), depending on the darkness. This applies to the four images in this section, changing just the meaning of the colour scale.

In Fig. 8.5, the Y-axis represents the sections (roman numbers), and the number of paragraph for each of them (arabic numbers). Section I comprises all the content before the judgment itself, including information such as the name of the case or the members of the Chamber. Since it is not titled, here it will be named “INTRODUCTION”. Section II is the “PROCEDURE”, usually short, where there is only one event in the first paragraph, corresponding to the event of “lodge an application” that originated the case under judgment. Section III is “THE FACTS” and, as can be appreciated in

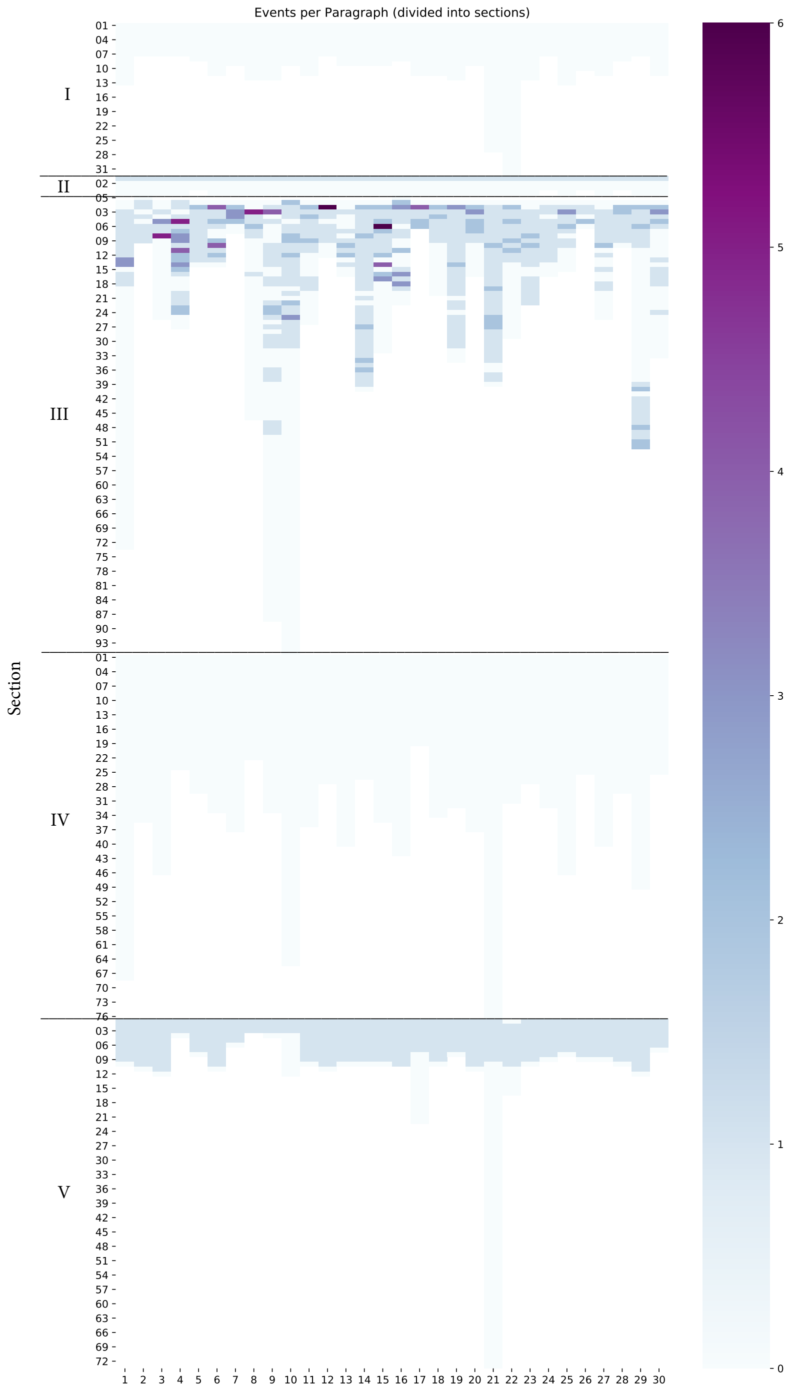


Figure 8.5: Events per Paragraph in the documents in EventsMatter corpus.

the figure, contains most of the events, distributed heterogeneously through the section. Due to this, Fig. 8.6 reproduces in more detail this section, and there can be seen that the amount of events and their distribution is not necessarily related to the length of the section; more paragraphs do not imply more events. Section IV, "THE LAW", contains no events, since it refers to the European and national legislation to which the case is related, citing it along with other merits and pertinent considerations. Finally, Section V includes the "FINAL DECISION" by the court, always following the structure:

> FOR THESE REASONS, THE COURT, UNANIMOUSLY,
>
> 1. {Decision I}
> 2. {Decision II}
> 3. {...}
>
> {Information about the date and language of the writing, along with the signatures and any annex attached.}

Fig. 8.7a represents the amount of events in each of the five sections previously described. The "INTRODUCTION" section, as already pointed out, has always one event, while "THE FACTS" presents a very variable amount of them, reaching in some cases forty events. This might be attributed to the different lengths of the section in each of the judgments, but Fig. 8.7b, showing the average events per paragraph on each section, belies it. Finally, the "FINAL DECISION" section is very uniform, except for some documents that present annexes or have longer sections for other reasons.

8.2.2 Training Strategies

Regarding the training strategy of the system, both semantic and syntactic considerations were used. On the one hand, all the events and attached arguments annotated in the training set of the EventsMatter corpus (presented in Section 7.1) were collected, and both the core of the events and the relations among their different parts were stored. On the other hand, also an external semantic resource, FrameNet, was used to enrich the keywords used to identify legal events. Subsequent sections provide a detailed description of both approaches.

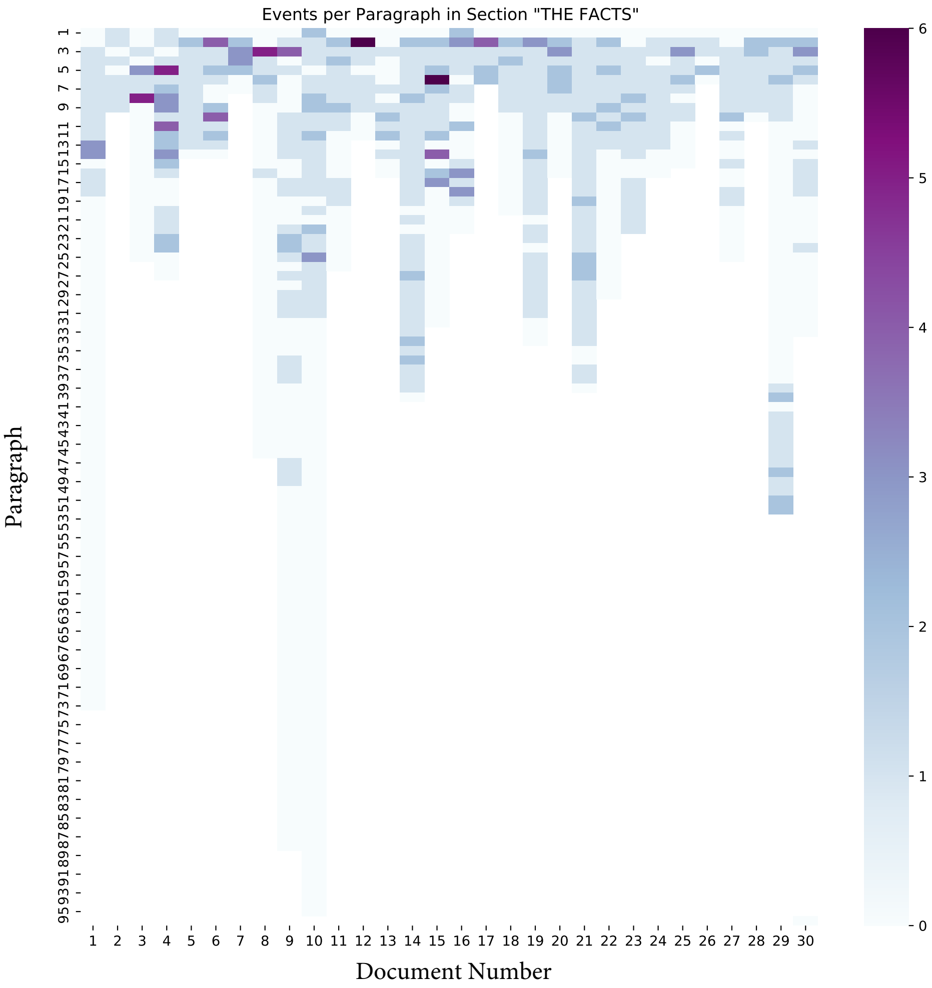
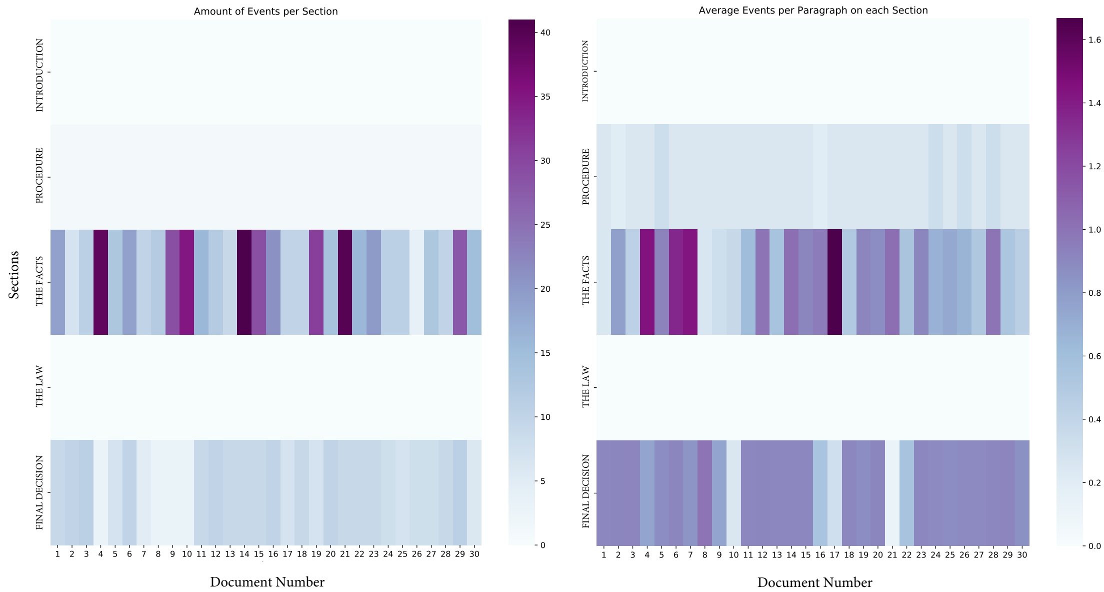


Figure 8.6: Amount of Events in the section "THE FACTS" in documents in EventsMatter corpus.



(a) Amount of events per section.

(b) Average events per paragraph.

Figure 8.7: Amount of events (a) and average events per paragraph (b) per Section in the documents in EventsMatter corpus.

8.2.2.1 EventsMatter Training Set

The first step of the training phase is to collect all the event mentions in the corpus training set. The parts of the sentences annotated as core are then isolated, and a sentence just with it, adding as generic subject “They” in order to make them simple to parse and grammatically correct, is generated. Thereupon a frame for each of the main verbs of these *simple* sentences that stores the information of all the mentions of these verbs along the corpus is created. This is, that for instance the verb “lodge” (that is to some extent a *light verb*[145] in the legal domain) can appear in several sentences carrying different meanings depending on the object attached. Some examples of its use would be the constructions “lodge a complaint”, “lodge a request”, “lodge an appeal”, “lodge an objection” or “lodge an action”. It should be noted that most of these cases could be simplified using a single semantic-carrying verb, such as “to complain” or “to request”, but that the legal domain tends to recur to these paraphrasing in texts, since they

[145] *Light verbs* are those verbs that have little semantic meaning, needing, therefore, more words to constitute a full predicate. This is for instance the case of the verbs “make” or “take” in English. For more information on this linguistic phenomenon, please check the work by Butt (2010).

usually imply not just an action but also a formal procedure (usually administrative).

The verbs found in this phase are outlined in Fig. 8.8, where their types and frequency are also presented. Each of them constitutes a *frame* that will be used to identify and classify future mentions of each of the verbs in new texts. The structure of the Frame class used to store the information gathered for each of these verbs, along with an example of the mentions and information collected for a specific verb, are depicted in Fig. 8.9.

Finally, it must be noted that, as shown in Fig. 8.9, a distinction between passive and active voice is made when searching for the dependency parsing relations among the members of the core of an event. This is a consideration that might not be important in general texts, but the legal domain tends to present a high rate of passive verbs. Among the events in the training set, for instance, 14% of the mentions were expressed as passive sentences.

Two couples of txt files containing (1) the *simple* version of each sentence with a relevant event mention and (2) the type of events of each of the mention are available within the system -a couple for all the sentences of the corpus (named *all*) and another for just the training part (*train*). The collection of events can be easily extended by adding to the files new sentences and their respective types, and then executing the respective main class in the system that creates a *events.ser* (*serialized*) file. This serialized file contains a HashMap of all events and their information in the form of Frames (the structure detailed in Fig. 8.9).

8.2.2.2 FrameNet training

It is straightforward that some events not present in the training set of the EventsMatter corpus should be detected in other documents, and even that events considered not relevant in those documents can be relevant in other cases.

This is why, in addition to the events gathered from the training set explained previously, it was decided to enrich the system with frames from FrameNet (Baker et al., 1998). FrameNet is a database that contains semantic frames together with the words that represent them in text, as well as additional information such as the arguments this frame can present. Since frames represent situations, they can be understood as events to some extent, and incorporating a selection of them to the target events would help to generalize the used approach.

VERB	OCCUR	TYPE	VERB	OCCUR	TYPE	VERB	OCCUR	TYPE	VERB	OCCUR	TYPE
lodge	42	36%	extend	3	100%	note	2	0%	kill	1	100%
uphold	19	37%	provide	3	33%	overturn	2	50%	appoint	1	100%
dismiss	18	39%	indicate	3	33%	admit	2	0%	interview	1	0%
ask	17	35%	place	3	67%	hear	2	0%	open	1	100%
have	15	53%	question	3	100%	bury	2	50%	discuss	1	100%
appeal	13	62%	die	3	100%	summon	1	100%	declare	1	0%
refuse	12	42%	exclude	3	67%	instigate	1	0%	attend	1	0%
find	12	75%	carry	3	0%	commit	1	0%	buy	1	0%
order	11	64%	allow	3	0%	attempt	1	100%	plead	1	100%
issue	11	55%	request	3	67%	put	1	100%	undertake	1	100%
apply	10	40%	publish	3	67%	cover	1	100%	privatise	1	100%
give	9	22%	challenge	3	67%	review	1	0%	fine	1	100%
quash	9	44%	respond	3	33%	deprive	1	0%	leave	1	0%
institute	8	63%	release	2	50%	reduce	1	0%	contact	1	100%
discontinue	8	50%	decline	2	50%	pass	1	100%	claim	1	0%
inform	7	14%	enter	2	0%	remain	1	100%	bear	1	0%
bring	7	57%	sentence	2	100%	agree	1	100%	vacate	1	100%
authorise	7	57%	fail	2	100%	drink	1	0%	consider	1	100%
impose	6	33%	oppose	2	0%	stop	1	0%	amend	1	0%
reject	6	50%	become	2	0%	detain	1	100%	telephone	1	100%
start	6	33%	exercise	2	0%	terminate	1	0%	duplicate	1	0%
marry	5	40%	seek	2	0%	begin	1	100%	complain	1	100%
undergo	5	20%	file	2	50%	object	1	100%	decrease	1	0%
return	5	40%	receive	2	0%	examine	1	0%	keep	1	0%
send	5	60%	learn	2	100%	seize	1	100%	exchange	1	0%
submit	5	40%	stay	2	0%	settle	1	100%	try	1	0%
grant	5	40%	report	2	50%	deliver	1	0%	occupy	1	0%
decide	4	50%	invite	2	50%	dissolve	1	0%	rule	1	100%
conclude	4	50%	arrest	2	50%	speak	1	0%	delete	1	100%
divorce	4	75%	sign	2	0%	convict	1	0%	make	1	0%
register	4	75%	hold	2	50%	acquit	1	0%	restore	1	0%
do	4	25%	initiate	2	50%	charge	1	100%	identify	1	0%
reply	4	25%	suspend	2	100%	set	1	0%	perform	1	100%
move	4	50%	establish	2	50%	forward	1	0%	go	1	0%
state	3	100%	take	2	0%	launch	1	0%	invalidate	1	0%
write	3	33%	transfer	2	0%	draw	1	0%	pronounce	1	0%
accept	3	33%	reopen	2	100%	suspect	1	0%	visit	1	100%

Figure 8.8: Events extracted from the EventsMatter training corpus. The second column (*OCCUR*) presents the amount of times that verb was annotated as relevant event. The third column (*TYPE*) shows the percentage of times it was typed as a *procedure* event (being the complementary percentage corresponding to the *circumstance* type).

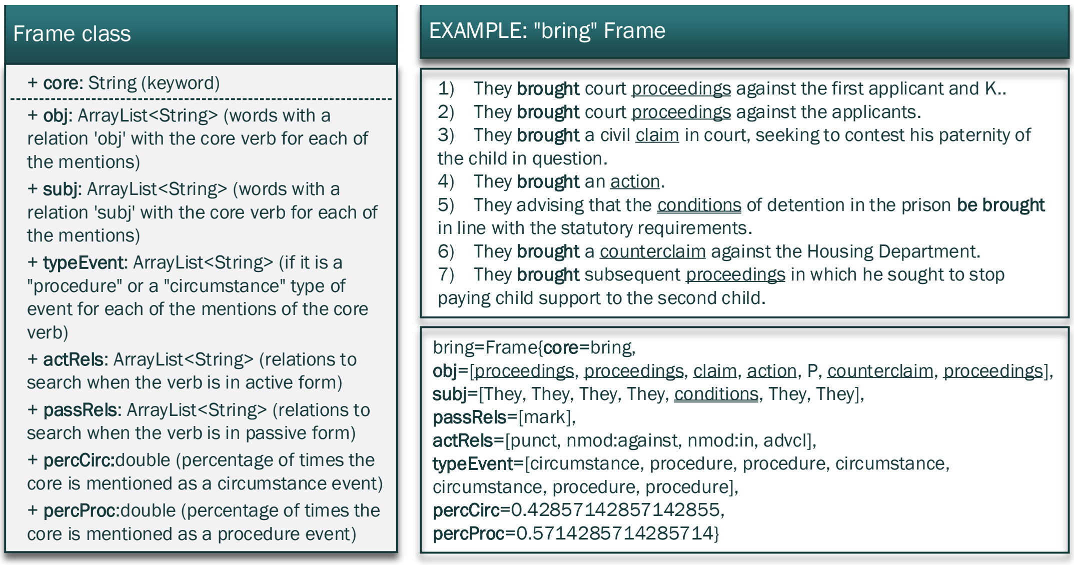


Figure 8.9: Frame Class to store the events in WhenTheFact (left side) and example with the verb "bring" (right side). In this case, seven different mentions of this verb were found in the corpus (top right), where the mention of the verb was marked in bold, the object was underlined and the subject in the case of passive voice was double underlined. Finally, the text box in the bottom-right shows how would be the frame extracted from these seven sentences. There it can be observed the different objects (obj) found (*proceedings, claim, action, counterclaim*), as well as a P in the fifth position of the array, meaning that that sentence was passive. In the **passRels** and **actRels**, the relations that connect the different parts of the core in the dependency parsing of the sentences (**passRels**, passive relations, from the 5th sentence, and **actRels** from the rest of them) can be found. Regarding **typeEvent**, it stores the different types of event (*circumstance* or *procedure*) the verb "bring") plays on each of the sentences. Finally, the percentage of these types is stored in the fields **percCirc** and **percProc**, that will help to decide if a mention found in a text is of one type or the other.

Since not all the frames in FrameNet are of interest, the database was manually inspected using the FrameGrapher tool[146], which allowed us to navigate through it and find the most relevant ones to the task. After examining the different relations among the frames, the most general ones were found, as well as their children, and their information was imported using a Python script and the library NLTK (Loper and Bird, 2002), including *framenet*. These most legally representative parent frames were namely "Committing_crime", "Crime_scenario", "Law", "Obligation_scenario", and "Misdeed". The frames collected from them, together with the lexical units associated with them (that is what will be looked for in the text), are detailed in Table 8.1. The non-lexical frames (this is, those that have no lexical units associated), in this case "Crime_scenario" and "Obligation_scenario", are not shown in the table for space purposes.

Table 8.1: Final selection of legal-related frames from FrameNet used in WhenTheFact.

Frame	**Lexical Unit (pos)**
Abusing	'abuse (n)', 'abuse (v)', 'abusive (a)', 'batter (v)', 'domestic violence (n)', 'maltreat (v)', 'maltreatment (n)'
Kidnapping	'kidnap (v)', 'abduct (v)', 'shanghai (v)', 'nab (v)', 'snatch (v)', 'kidnapping (n)', 'abduction (n)', 'kidnapper (n)', 'abductor (n)', 'snatcher (n)', 'kidnapped (a)', 'abducted (a)'
Piracy	'hijack (v)', 'hijacking (n)', 'hijacker (n)', 'carjacking (n)', 'hijacked (a)', 'piracy (n)', 'pirate (v)', 'carjack (v)'
Rape	'rape (v)', 'rape (n)', 'rapist (n)', 'raped (a)', 'sexually assault (v)'
Robbery	'rob (v)', 'robber (n)', 'mug (v)', 'robbery (n)', 'mugger (n)', 'mugging (n)', 'stick-up (n)', 'hold-up (n)', 'hold up (v)', 'rob blind (v)', 'stick up (v)', 'ransack (v)', 'rifle (v)'
Smuggling	'smuggle (v)', 'smuggling (n)', 'smuggler (n)', 'contraband (a)', 'contraband (n)'
Theft	'steal (v)', 'purloin (v)', 'filch (v)', 'snitch (v)', 'pilfer (v)', 'swipe (v)', 'lift (v)', 'pinch (v)', 'thieve (v)', 'thief (n)', 'pickpocket (n)', 'cutpurse (n)', 'pilferer (n)', 'snatcher (n)', 'theft (n)', 'thieving (n)', 'pilferage (n)', 'light-fingered (a)', 'thieving (a)', 'snatch (v)', 'nick (v)', 'embezzle (v)', 'misappropriate (v)', 'shoplift (v)', 'stealer (n)', 'shoplifter (n)', 'shoplifting (n)', 'pilfering (n)', 'stolen (a)', 'embezzlement (n)', 'embezzler (n)', 'peculation (n)', 'misappropriation (n)', 'larceny (n)', 'snatch (n)', 'stealing (n)', 'pickpocket (v)', 'heist (n)', 'flog (v)', 'abstract (v)', 'cop (v)', 'rustle (v)', 'bag (v)', 'abstraction (n)', 'make off (with) (v)', 'abscond (with) (v)'
Committing crime	'commit (v)', 'perpetrate (v)', 'crime (n)', 'commission (n)'
Offenses	'assault (n)', 'murder (n)', 'statutory rape (n)', 'sabotage (n)', 'manslaughter (n)', 'hijacking (n)', 'theft (n)', 'burglary (n)', 'robbery (n)', 'conspiracy (n)', 'larceny (n)', 'copyright infringement (n)', 'negligence (n)', 'possession (n)', 'felony (n)', 'sexual harassment (n)', 'treason (n)', 'battery (n)', 'kidnapping (n)', 'fraud (n)', 'indecent assault (n)', 'sexual assault (n)', 'child abuse (n)', 'homicide (n)', 'arson (n)', 'rape (n)'

[146] `https://framenet.icsi.berkeley.edu/fndrupal/FrameGrapher`

Table 8.1: (cont) Final selection of legal-related frames from FrameNet used in WhenTheFact.

Frame	Lexical Unit (pos)
Criminal investigation	'inquiry (n)', 'probe (n)', 'investigate (v)', 'inquire (v)', 'probe (v)', 'investigation (n)', 'lead (n)', 'clue (n)', 'case (n)'
Arson	'arson (n)', 'arsonist (n)'
Severity of offense	'actionable (a)', 'capital (a)', 'indictable (a)', 'felonious (a)'
Suspicion	'suspect (v)', 'under suspicion (of) (prep)', 'suspect (n)'
Arraignment	'arraign (v)', 'arraignment (n)'
Arrest	'arrest (v)', 'apprehend (v)', 'bust (v)', 'nab (v)', 'collar (v)', 'cop (v)', 'arrest (n)', 'bust (n)', 'apprehension (n)', 'book (v)', 'summons (v)'
Sentencing	'sentence (v)', 'sentence (n)', 'order (v)', 'send up (v)', 'condemn (v)'
Trial	'trial (n)', 'case (n)'
Appeal	'appeal (n)', 'appeal (v)', 'appellate (a)', 'appellant (n)'
Bail decision	'set (v)', 'fix (v)', 'order (v)', 'bail (n)', 'bond (n)'
Entering of plea	'plead (v)', 'plea (n)'
Notification of charges	'charge (v)', 'charge (n)', 'indict (v)', 'indictment (n)', 'accuse (v)'
Surrendering	'surrender (v)', 'turn in (v)', 'give up (v)', 'surrender (n)'
Court examination	'examine (v)', 'cross-examine (v)', 'cross (n)', 'cross-examination (n)', 'examination (n)'
Jury deliberation	'deliberation (n)', 'deliberate (v)'
Verdict	'pronounce (v)', 'find (v)', 'finding (n)', 'ruling (n)', 'convict (v)', 'conviction (n)', 'acquit (v)', 'acquittal (n)', 'verdict (n)', 'clear (v)', 'guilty (a)', 'not guilty (a)'
Law	'law (n)', 'code (n)', 'protocol (n)', 'act (n)', 'statute (n)', 'regulation (n)', 'regime (n)', 'policy (n)', 'order (n)'
Legality	'illegal (a)', 'legal (a)', 'lawful (a)', 'unlawful (a)', 'wrongful (a)', 'illicit (a)', 'licit (a)', 'permissible (a)', 'wrongly (adv)', 'wrong (a)', 'prohibited (a)', 'legitimate (a)', 'fair (a)', 'criminal (a)'
Prohibiting or licensing	'ban (v)', 'forbid (v)', 'prohibit (v)', 'proscribe (v)', 'outlaw (v)', 'ban (n)', 'prohibition (n)', 'bar (v)', 'allow (v)', 'entitle (v)', 'permit (v)', 'sanction (v)'
Being in effect	'effective (a)', 'effect (n)', 'force (n)', 'valid (a)', 'void (a)', 'null (a)', 'binding (a)'
Compliance	'adhere (v)', 'comply (v)', 'observe (v)', 'adherence (n)', 'compliance (n)', 'follow (v)', 'observance (n)', 'break (v)', 'violate (v)', 'contravene (v)', 'breach (v)', 'violation (n)', 'contravention (n)', 'breach (n)', 'flout (v)', 'conform (v)', 'obey (v)', 'compliant (a)', 'transgress (v)', 'transgression (n)', 'lawless (a)', 'contrary (a)', 'conformity (n)', 'keep (v)', 'honor (v)', 'abide (by) (v)', 'obedient (a)', 'observant (a)', 'play by the rules (v)', 'circumvent (v)', 'noncompliance (n)', '(in/out of) line (n)', 'disobey (v)', 'in accordance (a)', 'by-pass (v)'
Documents	'visa (n)', 'passport (n)', 'subpoena (n)', 'warrant (n)', 'certificate (n)', 'papers (n)', 'license (n)', 'summons (n)', 'diploma (n)', 'deed (n)', 'lease (n)', 'agreement (n)', 'treaty (n)', 'charter (n)', 'authorization (n)', 'deposition (n)', 'brief (n)', 'writ (n)', 'affidavit (n)', 'will (n)', 'testimony (n)', 'testament (n)', 'ruling (n)', 'finding (n)', 'opinion (n)', 'title (n)', 'orders (n)', 'contract (n)', 'permit (n)', 'document (n)', 'contractual (a)', 'accord (n)', 'confirmation (n)', 'identification (n)', 'business card (n)'
Enforcing	'enforce (v)', 'enforcement (n)'
Strictness	'authoritarian (a)', 'indulgent (a)', 'lenient (a)', 'liberal (a)', 'strict (a)', 'tolerant (a)', 'severe (a)'

Table 8.1: (cont) Final selection of legal-related frames from FrameNet used in WhenTheFact.

Frame	Lexical Unit (pos)
Giving in	'relent (v)', 'acquiesce (v)', 'yield (v)', 'cave in (v)', 'give in (v)', 'give way (v)', 'capitulate (v)', 'fold (to demands)' (v)', 'cave (v)', 'submit (v)'
Terms of agreement	'condition (n)', 'stipulation (n)', 'provision (n)', 'clause (n)', 'term (n)', 'parameter (n)'
Misdeed	'misdeed (n)', 'sin (v)', 'sin (n)', 'transgress (v)', 'transgression (n)', 'peccadillo (n)'
Guilt or innocence	'guilty (a)', 'innocent (a)', 'guilt (n)', 'innocence (n)', 'blood on hands (n)'

A txt file containing all this information is available in the system. In order to add more frames, it is just needed to add them to the file maintaining the same format. The system has a main class named *readFrames* that will generate a frames.ser file from it, and is this file that is read by the system in order to facilitate its later use, storing the information in the form of a HashMap of structures containing the name, the core and the POS tagging information.

8.2.3 Event Extraction

Regarding the event extraction itself, Fig. 8.10 depicts the pipeline of the tool. The different stages of the processing below are detailed below.

The first step consists of finding the relevant parts of the text to annotate, using for this the Structure Extractor detailed in Section 8.2.1. If the structure is not recognized, the whole text will be annotated, which obviously impacts negatively in the amount and quality of the events. Otherwise, just the relevant parts of the document are processed subsequently.

The next step is to find the sentences involving temporal expressions. To this aim, the functionality of Añotador (Navas-Loro and Rodríguez-Doncel, 2020), the temporal tagger presented in Section 6.1, was adapted and integrated. If there is at least one temporal expression in a sentence, it is checked if it is a special case (namely the application lodgement, which always follows the same syntactic structure). If so, the arguments are annotated and the system goes to the next sentence. If not, it is checked if the sentence contains any of the events stored in events.ser, which contains the information gathered from the training corpus. If so, it is checked if the dependency parsing (*deppar*) of the sentence (using CoreNLP Manning et al. (2014)) is valid and the arguments (see (1) below) are detected. If not, it is checked again for the frames stored in frames.ser (the

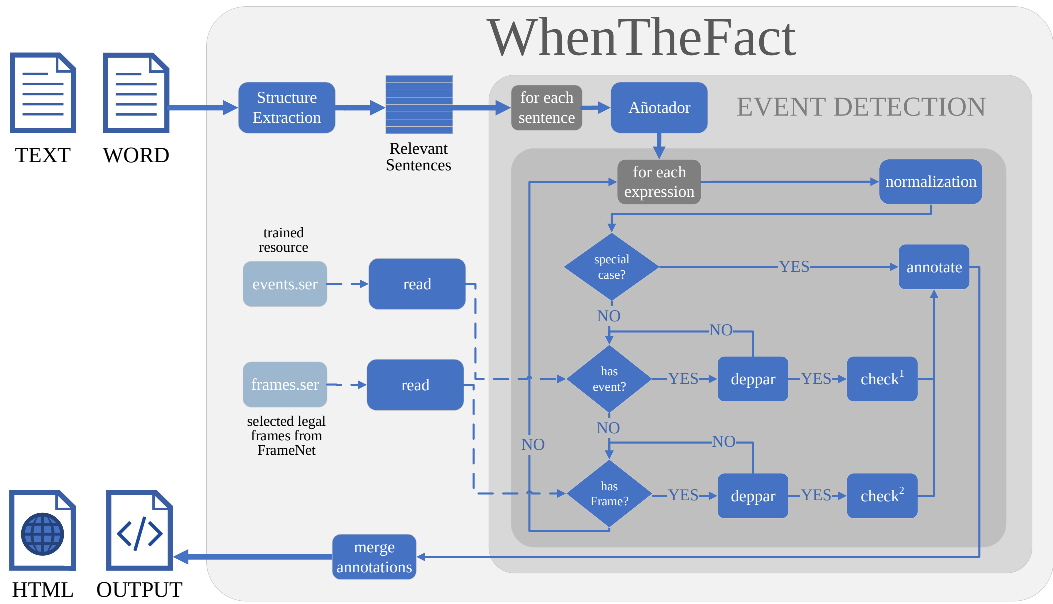


Figure 8.10: Pipeline of WhenTheFact.

legal frames specifically selected from FrameNet), or any word considered relevant (to do this, the first part of the sentence, usually including keywords, is analyzed, trying to detect semantic similar words in the body of the judgment). If this is the case, they are checked similarly to in the case of the events (see (2)). Once the main event is detected in the sentence, if there was more than one temporal expression in it, the temporal expression that is the closest to the core of the event will be selected.

(1) For the events, it is checked if it is not an auxiliary verb and if it is not in the gerund form. Then it is checked if it is in passive or active voice. Depending on this, either the relations stored in events.ser gathered from passive training cases or from active cases will be used.

(2) For the frames, the check function is similar to the events' one, but there are no specific relations stored for each frame, so the argument "who" and the extent of the core are therefore detected using default relations.

Once all the sentences have been explored, all the annotations are merged and the output is produced. This output consists of an annotated xml and as a visual HTML that also includes a timeline built from the retrieved events, as can be seen in Fig. 8.11.

8.2.4 Evaluation

The results of the last version of WhenTheFact over the EventsMatter corpus have verified its improvement with regard to its first implementation (the one reported in the related publication (Filtz et al., 2020)). The evaluation is depicted in Table 8.2.

		Event				**Event Components**					
		Identification		Type		What		When		Who	
		Len	Str	Len	Str	Len	Str	Len	Str	Len	Str
OLD	P	**0.86**	**0.80**	**0.47**	**0.43**	**0.80**	**0.24**	0.78	0.72	**0.75**	**0.69**
	R	0.78	0.73	0.43	0.39	0.69	0.21	0.63	0.59	0.63	0.58
	F	**0.82**	**0.76**	**0.45**	**0.41**	**0.74**	**0.22**	0.70	0.65	**0.69**	**0.63**
NEW	P	0.75	0.70	0.40	0.36	0.67	0.18	**0.80**	**0.76**	0.61	0.55
	R	**0.85**	**0.79**	**0.46**	**0.41**	**0.79**	**0.22**	**0.84**	**0.79**	**0.74**	**0.68**
	F	0.80	0.75	0.43	0.39	0.73	0.20	**0.82**	**0.78**	0.67	0.61

Table 8.2: Comparison between the previous implementation of the WhenTheFact event extractor (OLD) and the new implementation (NEW).

All the results are provided both *Lenient* (Len) and *Strict* (Str). Lenient results mean that the annotation is counted correct even it does not exactly match the reference annotation against which it is compared, but at least overlaps it. If for instance the correct annotation was "upheld the judgment" but just "upheld" was annotated, lenient

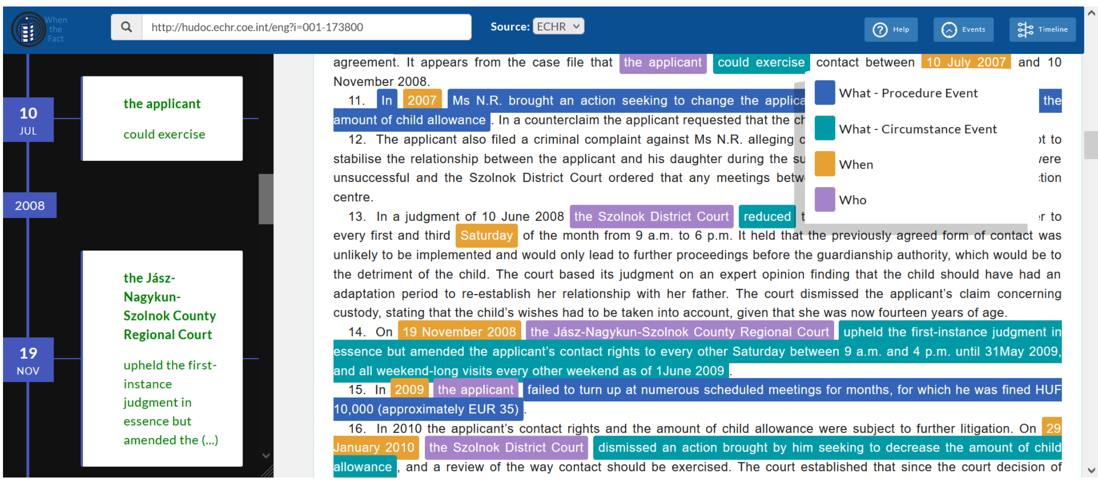


Figure 8.11: Screenshot of the demo of WhenTheFact, with the timeline created on the left and the annotated document on the right.

measures would consider the annotation correct. On the opposite, strict annotations just count as positive annotations that match exactly the extent of the reference annotation. As metrics, *Precision, Recall* and *F-Measure*, which are the standard measures used for evaluation of NLP systems, are used. Finally, in Table 8.2 the results for event identification (*Event* tag from the EventsMatter format, two first columns), for type classification (*circumstance* and *procedure*, columns three and four), and the extension detection for each of the arguments of the events, namely *Event_what*, *Event_when* and *Event_who*, are presented.

Although the results reported in the EventsMatter corpus paper (Filtz et al., 2020) might look better than the ones presented here, several issues must be taken into account. First, that the WhenTheFact approach processes the full text and detects the events, while the deep learning approaches results correspond to the annotation of sentences where it was already known that there was an event. Second, that the first implementation of WhenTheFact (the one included in the paper) was just conceived for ECHR documents. The last version is much more general, including more training and frames, and is, therefore, able to cover more types of documents, which explains the drop in precision. Nevertheless, in the legal domain is much more preferable to find all the relevant events and mark some extra ones than to miss some by trying to be very precise (this is, recall is more important than precision), so the new figures should be considered an improvement.

Summary

In this chapter two tools able to extract events from texts were presented. The first one, ContractFrames, is able to detect events related to the lifecycle of real-estate contracts from raw texts. The second one, WhenTheFact, identifies relevant events from European judgments. It is able to extract the structure of the document, as well as when the event happened and who carried it out, and it builds a timeline that allows the user to navigate through the annotations in the document. This confirms hypothesis H1.b, H2.a and H3 of the thesis.

The next chapter will present different tools that help to represent the events extracted from a legal decision in a semantic format that will populate an event-based knowledge graph.

Chapter 9

Event-related resources

In the State of the Art from Chapter 2, it was exposed that there is a gap of resources and interoperability in the event annotation domain. This chapter will present some of the contributions produced during this thesis in order to cover this gap.

The previous chapter presented an event extractor able to detect events in European court decisions, assuming that a judgment can be described as a series of time-marked happenings called *events* instead of focusing on the other entities. It would be now desirable to be able to represent this information in a semantic fashion that can be easily exploited for further tasks.

Therefore, once the events from the documents are extracted, it would be desirable to be able to represent them in a semantic format (e.g. as an ontology). In order to facilitate their retrieval, a knowledge graph focused on events could be built. Finally, taking into consideration that the legal domain practitioners are not usually familiar with semantic web technologies, a service with a series of predefined queries in order to facilitate consulting this knowledge graph is provided.

In order to fulfil all these steps, the following resources have been created as a complement to the event extractor WhenTheFact presented in Chapter 8:

a) an ontology supporting the representation of temporal information, which eases the translation between time-related formats[147],

b) a converter that takes temporal annotations in various forms and outputs them as RDF[148],

[147] `https://fromtimetotime.linkeddata.es/ontology.html`
[148] `https://fromtimetotime.linkeddata.es/service.html`

b) an event-based knowledge graph of legal judgments in English that can be easily queried[149].

This chapter presents the aforementioned contributions. Section 9.1 presents the ontology built to represent events and temporal information, while Section 9.2 covers the translation tool created to do the transition among formats. Finally, Section 9.3 presents the event-based knowledge graph generated from the previous tools, as well as possible exploitation options. In order to facilitate testing the interaction of all these contributions (along with the event extractor), a webpage that allows testing the pipeline step by step[150] has been created.

9.1 fromTimeToTime Ontology

In order to properly represent the temporal information extracted from the documents, an ontology named fromTimeToTime (ft3) has been created. The purpose of this ontology is double-folded: on the one hand, to be able to represent information from the annotations related to time and events that the current ontologies do not cover. On the other hand, to facilitate the translation between one annotation format or temporal representation option to another.

This section will briefly introduce this new ontology, stressing the main design decisions. The later section, which will describe the format converter, will also present some examples of the expected use of the ontology.

9.1.1 Ambitions

The objective of the ontology is to represent the temporal information present in the text, harmonizing existing non-ontological standards, adding relevant information not included in those and also alternative time-related information representation from other ontologies, as described in the Ontology Requirement Specification Document in Annex D. The scope is in brief representing temporal information annotation and events to facilitate the event-based representation of a document, stressing the temporal information annotation information and allowing different formats of temporal information. The pursued goals are therefore the following:

[149] `https://fromtimetotime.linkeddata.es/sparql.html`
[150] `https://fromtimetotime.linkeddata.es/pipeline.html`

1. Event-centric representation of information (in the form of an event-based knowledge graph, for instance).
2. Facilitate translation among annotation formats and time-related ontologies.
3. Storage of annotations for latter tasks (e.g. visualization, search).
4. Representation of events for different uses (e.g. timeline generation, knowledge graphs, pattern recognition).

Main expected users of this ontology are people involved in temporal tagging, or in NLP in general, but also people in need of representation of temporal information, especially for LinkedData or Knowledge Graph building.

9.1.2 LOT Methodology

For the ontology creation, the LOT methodology (Poveda-Villalón et al., 2019) was followed. The steps of this methodology are presented below together with a description of the work carried out at each stage.

1. **Ontology Requirements specification**: this stage encompass several activities, such as the use case specification, the scope definition and the collection of the requirements. The main output of this step is the Ontology Requirement Specification Document, included in Annex D, that summarizes the intended uses and requirements of the ontology.
2. **Ontology implementation**: this was the most difficult part of the development, since it included the search of related ontologies and the modelling itself. The Chowlk visual notation[151] was used during this phase in order to build a diagram representation, later transformed to OWL by the Code Generation Service. Finally, evaluation was checked using the OOPS Pitfall Scanner (Poveda-Villalón et al., 2014).
3. **Ontology publication**: Once the ontology was ready, again tools from the Ontology Engineering Group were used to publish it. WIDOCO (Garijo, 2017) helped in the generation of the documentation, available on the website of the ontology.

[151] https://chowlk.linkeddata.es/

4. **Ontology maintenance**: Regarding maintenance, an email to report bugs is available with the ontology. Also, maximum dissemination has been carried out to promote its use, detect possible shortcomings and fine-tune its usability.

9.1.3 Ontology design decisions

Although some previous ontologies mentioned in literature have covered temporal annotations to some extent, most of them are no longer available[152], so they could not be reused to this aim.

Several studies have analyzed the suitability of existing time-related ontologies for real-world tasks. For instance, Chen et al. (2018) highlight that the TEO ontology, although it allows the representation of periodic repetitions (SETs according to TimeML terminology), does not cover irregular repetitions of events. Additionally, the modelling they propose for periodic ones is sustained on durations, not covering for instance "Every October". In the modelling of the ontology, it has been tried to tackle these gaps. Still, reusing ontologies has also been maximized, prioritizing among them the W3C Time ontology, since it is the most well-known and stable one.

Regarding event representation from the point of view of mentions in text and relation with realizations and time anchoring, several possibilities are distinguished. Let us define first some concepts for the sake of understanding and clarification. Following TimeML nomenclature, there are *event instances* and *event tokens. Event instances* are realizations of the events, including information such as cardinality or polarity. On the other hand, *event tokens* are the mentions of events in a text. Although the relation between *event tokens* and *event instances* is usually one-to-one, corresponding one token to one instance, this is not always the case. After a thorough analysis, several possible combinations were detected among event instances, event tokens, temporal expressions and other arguments of the events, as can be seen in Table 9.1.

Some of these especially tough issues in representation are explained in more detail in the following sections.

9.1.3.1 Temporal Expression representation

One of the objectives of the ontology was to be able to represent any time-related annotation format. Due to this, some high-level classes were created, namely *Guidelines*,

[152] e.g. `http://www.newsreader-project.eu/ns/NAACL2013/id_1_eecb_txp_txt_xml_`

Annotation and *Argument*, that allow to create subclasses and instances for specific implementations. Additionally, also some abstract classes that allow unifying to some extent the different representations, such as the case of the class *Temporal Expression*, were added.

Different tags and concepts in the TimeML annotation standard were implemented as an exemplary, the most well-known annotation format for temporal expressions. Thus, the ontology offers, for example, the different arguments for the concepts considered in the annotation standard (temporal expressions, events, event instances and signals), with instances for the valid values of these arguments, but leaving the option of eventual extensions. These are, at the same time, related to other classes in the ontology, as the case of the class *temporal expression* shown in Fig. 9.1.

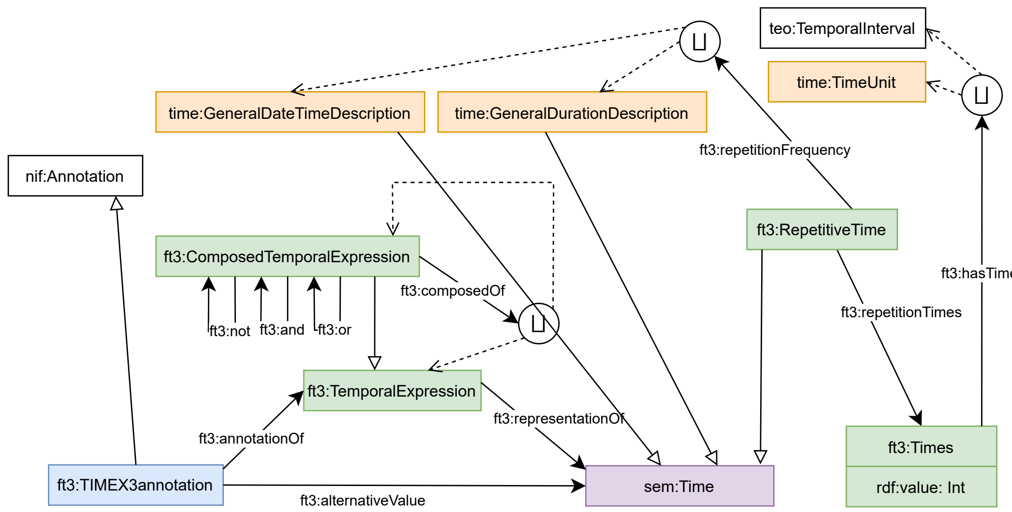


Figure 9.1: Excerpt of the ft3 ontology related to temporal expression representation.

Fig. 9.1 depicts the relation implemented in the ontology among the class *ft3:Temporal Expression*, the class *ft3:TIMEX3Annotation* and the class *sem:Time* from the Simple Event Model, used as abstract class to represent Time. This class is also linked to classes from the Time ontology, and can as well be associated with any other temporal representation option.

Additionally to the integration of these already existing representations, it was decided to add also the class *ft3:ComposedTemporalExpression* in order to be able to represent temporal expressions not currently covered by the existing standards. This class enables to join, intersect or negate a temporal expression, allowing to represent in a simple way complex expressions such as "*All days but Mondays*" or "*On Monday or Tuesday*".

9.1.3.2 Event representation

Regarding events, the main consideration to be represented in the ontology is the distinction about the following concepts:

- Event mention: the textual reference in the text. There can be several references to an event in a text (coreference). Also, a mention can be related to several events or subevents. This event mention can have attached an annotation.
- Event schematization: the abstract representation of the information about an event, such as *who*, *where*, and so on. It is a midpoint between text, reality and abstraction. This representation can be useful to support Question Answering (QA) routines.
- Event instance: the actual happening of an event in reality. One mention can imply several instances. Also, in some cases, the amount of instances cannot be derived. This concept is especially important for timeline building.
- Event formalization: it is an abstract representation of the event, a possible formalization in the form of frame, for instance. It can be considered as a way to classify events by linking them to resources such as WordNet or FrameNet.

Fig. 9.2 shows how these concepts were formalized in the ontology. Besides the four main concepts, TimeML event-related concepts MAKEINSTANCE and EVENT are associated to *ft3:EventInstance* and *ft3:EventMention*, respectively. Similarly, the event annotations from the EventsMatter corpus are also represented but linked to *ft3:EventSchematization*, since they provide information such as *who* and *when*. Finally, as happened with *sem:Time*, the event representation is related to the equivalent for event in the SEM ontology, *sem:Event*.

Furthermore, in order to clarify how these concepts reflect real annotations, Table 9.1 shows different examples of sentences and how they would comply with this representation. Some of these examples are discussed further below:

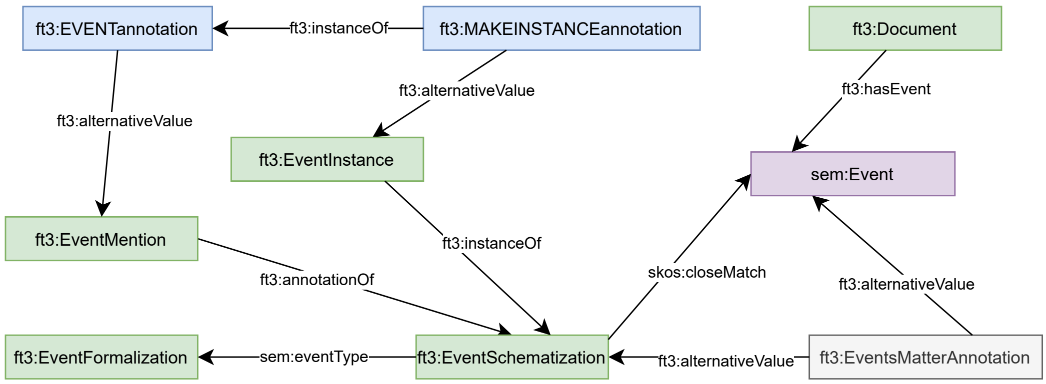


Figure 9.2: Excerpt of the ft3 ontology related to event representation.

#	Event	Sentence	TEx	Men	Sch	Ins
a	go	Yesterday I **went** to the park.	1	1	1	1
b	go	I **went** to the park on the 5th and the 6th.	2	1	1	2
c	go	I **went** to the park.	0	1	1	1
d	go	I **go** to the park every Tuesday.	1	1	1	X
e	go	I **went** to the park. During the **stroll**, it started raining.	1	2*	1	1
f	meet	They **met** several times.	1	1	1	X
g	concert	The **concert** was cancelled.	0	1	1	0
h	cancel	The concert was **cancelled**.	0	1	1	1
i	attend	The applicant **did not attend**.	0	1	1	0
j	skip	He **skipped** the sessions.	0	1	1	X
k	attend	He skipped the sessions.	0	1	1	0
l	sessions	He skipped the **sessions**.	0	1	1	X
m	admit	The appeal **was not admitted**.	0	1	1	1
n	refuse	The appeal **refused**.	0	1	1	1

Table 9.1: Example of sentences and the corresponding representation attributions. The first column shows the letter assigned to the example, while the second column presents the event in the spotlight and the third one the example sentence itself. The last four columns show the number of Temporal Expressions (TEx, underlined in the sentence), Event Mentions (Men, in bold in the sentence), Event Schematizations (Sch) and Instances (Ins) the sentence would produce. (*) The stroll has been considered a meronymic coreference of the event "go", but could also be considered a subevent.

a) This example is the simplest. One temporal expression and one mention of an event lead to a single schematization and a single instance.

b) In this case, there is still one mention of event and one schematization, but two temporal expressions associated; the action happens twice (one each day) and therefore there are two event instances.

c) In this example there is no temporal expression, but it can be assumed that the event happens once, so just one instance would be derived.

d) In this sentence, periodic temporal come into play. The only expression suggests that the event happens several times, but there is no clue about how many.

e) If "the stroll" is considered a mention of the event of going to the park, there is a coreference, and therefore just one mention.

f) This case is similar to case d), except for the fact that the temporal expression is not periodic, but simply implies more than one happening.

g) and h) Both examples share the sentence, but depend on which event is under the spotlight for its formalization. If the focus is on the concert (example g)), it did not happen, so there is no instance. On the contrary, in example h), the cancellation is an actual event, so there is one instance. How to decide how to interpret this situation will usually depend on the specific use the user is dealing with.

i) This is a very interesting example from the legal point of view. The fact that someone did not attend to a view or a trial is commonly reflected in judgments. Although the event of attending did not happen, so there is no instance of it, the following example shows similar cases expressed differently.

j) k) and l) Another way to express that someone did not attend a procedural event is to say they "skipped" it. Therefore, being the same case as i), the fact of no acting becomes an act itself and can have consequences.

m) and n) Here we find again the case of an event that can be both equally described with a verb or its negated opposite. Differently from the case of the concert, the fact of refusing or not admitting an appeal does not mean it does not happen: the appeal actually happened, and this is just the result of the deliberation on it. Therefore, here the negation is clearly still an event, because the fact of not admitting an appeal is an action itself, just expressed as the negation of one of the two possible results.

After these examples, the problematic existing between the different ways of understanding the same event depending on how it interacts with the temporal expressions or on the characteristics of the event itself become evident. There is not a correct way of understanding or representing events, and the meaning extremely depends on the situation and its particularities, the context of the case and the requirements of the use case for which the representation is needed.

Finally, in order to guarantee and facilitate the use of the ontology, it has been assessed with the OOPS! Ontology Pitfall Scanner (Poveda-Villalón et al., 2014) and documented using the WIDOCO wizard for ontology documentation (Garijo, 2017). The documentation (including evaluation[153], mainly consisting of minor comments and with no critical pitfalls) can be checked in the ontology webpage, where it is published together with the ontology itself. Both are additionally available in Zenodo[154].

9.2 FromTimetoTime Converter

One of the main missing tools identified dealing with time-related information is the gap existing between the task of finding temporal information in texts and its latter usage for further tasks. Besides the existence of many time-related ontologies and options, such as Temporal Description Logics in order to reason over them, there is no bridge between them and the pure NLP task.

In order to cover this need, a converter able to read different temporal annotation formats and output them in different formats, including the ontology previously presented, has been created. Regarding the implementation, the doctorate candidate is well aware that there are recommended tools for some of the tasks performed. For example, for the translation of TimeML XMLSchema to ontology format GRDDL[155] or an XSLT transformation could had been used. However, while the former is obsolete, the latter was overly complex and did not worth doing it manually. Additionally,

[153] `https://fromtimetotime.linkeddata.es/ontodoc/OOPSEvaluation/OOPSeval.html`
[154] `https://zenodo.org/record/5034640`
[155] `https://www.w3.org/TR/grddl/`

alternative representations to the ontological one could be used, such as the W3C Web Annotations recommendation[156].

This service is currently able to read TimeML and EventsMatter documents, as well as ft3 ontology documents, and transform them into the following formats:

- EventsMatter: TimeML documents or ft3-ontology formatted documents can be translated to the EventsMatter format. Fig. 9.3 shows an example of this format.
- TimeML: ft3-ontology formatted documents or in the EventMatters format can be translated to the TimeML standard. In Fig. 9.4 the TimeML output of the converter for the previously mentioned example can be found.
- ft3: the annotations of both annotation formats will be expressed in the form of the ft3 ontology. Fig. 9.5 presents the example introduced in Fig. 9.3 as ft3 RDF.
- ft3+time: additionally to the RDF representation of the annotations, the temporal expressions annotated will be transformed to time-related ontology data, mainly to the Time Ontology, but also to complementary ones from other ontologies. The addition in the case of the previous example is presented in Fig. 9.6.
- ft3+events: in addition to the RDF representation of the annotations, the events detected in the text are also represented as sem:Event classes. They contain the information of the arguments that might be annotated in the original text, such as *sem:hasActor* or *sem:hasTime*. The part that would be added for the example is presented in Fig. 9.7.

Besides these formats, it is also possible to extend the converter to include more options. In order to do so, a pivot class named MAP that can be considered an "interlingua" was implemented. This class is a map of Strings where the key is the identifier of the argument. In order to know how each type of annotation must be interpreted, when each annotation is read a *metatype* is assigned to it. For instance, both the *Event_when* tag (from EventsMatter format) and the *TIMEX3* one (from the TimeML standard) have the metatype *TEMPORAL ANNOTATION*, while *Event_what* and *EVENT* have *EVENT ANNOTATION*. Table 9.2 shows the correspondence of some of these metatypes, as well as the mapping among the arguments.

[156] `https://www.w3.org/TR/annotation-model/`

metatype	TEMPORAL		EVENT		WHO
MAP	TIMEX	Event_when	EVENT	Event_what	Event_who
TYPE	type		class	type	
ID	tid		eid	tid	tid
VALUE	value				
SENTID	sentid				
FUNCTIONINDOCUMENT	functionInDocument				
TEMPORALFUNCTION	temporalFunction				
VALUEFROMFUNCTION	valueFromFunction				
MOD	mod				
ANCHORTIMEID	anchorTimeID				
BEGINPOINT	beginPoint				
ENDPOINT	endPoint				
QUANT	quant				
FREQ	freq				
LEMMA				lemma	
STEM			stem		
PROV				prov	

Table 9.2: Correspondence among different annotations and MAP. Each of the values of the column map has a correspondent object property in the ft3 ontology (e.g., TYPE has ft3:hasType).

```
On <Event_when tid="t4" type="DATE" value="1990-10-06">6 October 1990</Event_
when> <Event_who argument="who" tid="t4">he</Event_who> <Event_what argument=
"what" tid="t4" type="circumstance" prov="eventsmattertrain" lemma="marry">ma
rried</Event_what> Ms N.R.
```

Figure 9.3: Example of text annotated in the EventsMatter format.

```
<?xml version="1.0" ?>
<TimeML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespace
SchemaLocation="http://timeml.org/timeMLdocs/TimeML_1.2.1.xsd">
On <TIMEX3 tid="t4" type="DATE" value="1990-10-06">6 October 1990</TIMEX3> he
<EVENT eid="t4" class="circumstance">married</EVENT> Ms N.R.
</TimeML>
```

Figure 9.4: Output of the converter as TimeML.

This MAP facilitates the task of translating among all the different formats. Consequently, to add a new format it will be necessary to simply perform the following

```
<https://fromtimetotime.linkeddata.es/doc/samples/doc002>
a nif:Context , ft3:Document ;
nif:beginIndex "0"^^xsd:nonNegativeInteger ;
nif:endIndex "36"^^xsd:nonNegativeInteger ;
nif:title    "X"^^xsd:String ;
nif:isString """On 6 October 1990 he married Ms N.R.""" ;
nif:AnnotationUnit [
    <https://fromtimetotime.linkeddata.es/doc/samples/doc002/EventsMatter/
    Event_whenannotation_t4_5> [
    a ft3:EventsMatterEvent_when ;
    nif:beginIndex "3"^^xsd:nonNegativeInteger ;
    nif:endIndex "17"^^xsd:nonNegativeInteger ;
    ft3:hasID "t4"^^xsd:String ;
    nif:isString """6 October 1990""" ;
    ft3:hasTid "t4"^^xsd:String;
    ft3:hasValue "1990-10-06"^^xsd:String;
    ft3:hasType ft3:DATE ;
   ];
    <https://fromtimetotime.linkeddata.es/doc/samples/doc002/EventsMatter/
    Event_whatannotation_t4_6> [
    a ft3:EventsMatterEvent_what ;
    nif:beginIndex "21"^^xsd:nonNegativeInteger ;
    nif:endIndex "28"^^xsd:nonNegativeInteger ;
    ft3:hasID "t4"^^xsd:String ;
    nif:isString """married""" ;
    ft3:hasType ft3:circumstance ;
    ft3:hasProv "eventsmattertrain"^^xsd:String;
    ft3:hasLemma "marry"^^xsd:String;
   ];
    <https://fromtimetotime.linkeddata.es/doc/samples/doc002/EventsMatter/
    Event_whoannotation_t4_7> [
    a ft3:EventsMatterEvent_who ;
    nif:beginIndex "18"^^xsd:nonNegativeInteger ;
    nif:endIndex "20"^^xsd:nonNegativeInteger ;
    ft3:hasID "t4"^^xsd:String ;
    nif:isString """he""" ;
   ];
] .
```

Figure 9.5: Output of the converter with the output format *ft3*. Prefixes are not included in order to avoid verbosity.

```
ft3:alternativeValue [
    <https://fromtimetotime.linkeddata.es/doc/samples/doc002/Time_t4> [
    a sem:Time,
    time:GeneralDateTimeDescription ;
    time:year "1990"^^xsd:gYear ;
    time:monthOfYear greg:October ;
    time:month "--10"^^xsd:gMonth ;
    time:day "---06"^^xsd:gDay ;];
    ];
];
```

Figure 9.6: Additional output of the converter with the output format *ft3+time*.

```
ft3:hasEvent [
<https://fromtimetotime.linkeddata.es/doc/samples/doc002/EVENT_t4> [
a sem:Event ;
sem:EventType "marry" ;
ft3:hasType ft3:circumstance ;
ft3:hasID """t4""" ;
sem:hasTime [
<https://fromtimetotime.linkeddata.es/doc/samples/doc002/Time_t4> [
    a sem:Time, time:GeneralDateTimeDescription ;
    time:year "1990"^^xsd:gYear ;
    time:monthOfYear greg:October ;
    time:month "--10"^^xsd:gMonth ;
    time:day "---06"^^xsd:gDay ; ];
] ;
sem:hasActor """he"""^^xsd:String ; ]
].
```

Figure 9.7: Additional output of the converter with the output format *ft3+events*.

steps:

- Create a new class that implements the “AbstractAnnotation” class for each new annotation and whose constructors receive MAP as an argument.
- Add a constructor to MAP that receives the new class.
- Create a reader of that format that stores the annotations in Document format.
- Add to Document an option to be translated to the new format.

Similarly, for handling the conversion of TimeML values to the ontology format (or to any other temporal format) another pivot map named TIMEMAP, detailed in Table 9.3, was used.

TYPE	DATE	DURATION	TIME	SET-DATE	SET-DURATION
TIMEUNIT				X	X
TIMEAMOUNT				X	X
REF	X				
YEAR	X	X	X	X	X
SEASON	X	X	X	X	X
WEEK	X	X	X	X	X
WEEKDAY	X	X	X	X	X
HALFYEAR	X	X		X	X
TRIMESTER	X	X		X	X
QUARTER	X	X		X	X
ERA	X				
DAY	X	X	X	X	X
MONTH	X	X	X	X	X
DECADE		X			X
CENTURY		X			X
MILLENIUM		X			X
SECOND		X	X		X
MINUTE		X	X		X
HOUR		X	X		X
PARTDAY			X		

Table 9.3: Correspondence between TIMEMAP keys and the information contained different types of temporal expressions in the TimeML standard. The SET type has been divided since its value can be in the form of a DATE or a DURATION.

In the case of DATEs or TIMEs, the information is represented as part of a *time:GeneralDateTimeDescription*. The correspondence of each value of the TIMEMAP to the Time ontology is therefore to properties such as *time:day*. Additionally, for temporal expressions not covered by the Time ontology, as mentioned before, the TEO ontology[21] and the INTERVALS resource[157] were used. This is the case of the key PARTDAY, that represents parts of the day such as *morning* or *noon*, where *teo:TEO_0000190* (labeled *Instant of the day*) was used to describe that property and its object (*teo:TEO_0000194* and *teo:TEO_0000195*, respectively). In the case the object was

[157] `http://reference.data.gov.uk/def/intervals/`

not available, one individual was created in out ontology (e.g. *ft3:NIGHT*). In other occasions, time had the property but not the right object, as in the case of *quarters, trimesters* or *semesters*, where INTERVALS were used. Finally, in some cases, such as references to the past, present or future (represented in TimeML as DATEs with values *PAST_REF*, *PRESENT_REF* and *FUTURE_REF*, respectively), also the property (*ft3:hasTimeRef*) was added.

On the other hand, in the case of DURATIONs, the information was represented as part of a *time:GeneralDurationDescription.* The Time ontology properties and objects were again prioritized, using for instance *time:days* or *time:years* to represent the amount of days and years in the DURATION.

Finally, SETs are described using a class with two different properties, namely *ft3:RepetitiveTime* and the properties *ft3:repetitionFrequency* and *ft3:repetitionTimes.* The first property would represent the frequency of a periodic event, while the second corresponds to the amount and granularity of the repetition. Fig. 9.8 and Fig. 9.9 represent the temporal information of the expression "*Twice a week*" and "*Three days every two months*", respectively.

```
ft3:alternativeValue [
 <https://fromtimetotime.linkeddata.es/ doc/samples/doc002/Time_t1> [
 a sem:Time, ft3:RepetitiveTime ;
ft3:repetitionFrequency [
     time:weeks         "1"^^xsd:decimal ;
    ];
ft3:repetitionTimes [
    ft3:hasTimeUnit    ft3:TIMES ;
    rdf:value         2^^xsd:nonNegativeInteger ;
    ];
];];
```

Figure 9.8: Alternative value of the temporal expression "*Twice a week*".

The code of the converter is available online[158] and can be freely adapted. The converter can also be tested in the fromTimeToTime webpage.

[158] https://github.com/mnavasloro/FromTimeToTime

```
ft3:alternativeValue [
 <https://fromtimetotime.linkeddata.es/ doc/samples/doc002/Time_t1> [
 a sem:Time, ft3:RepetitiveTime ;
ft3:repetitionFrequency [
     time:months         "2"^^xsd:decimal ;
    ];
ft3:repetitionTimes [
    ft3:hasTimeUnit    time:DAY ;
    rdf:value         3^^xsd:nonNegativeInteger ;
    ];
];];
```

Figure 9.9: Alternative value of the temporal expression "*Three days every two months*".

9.3 Legal Event-Based Knowledge Graph

Event-Centric Knowledge Graphs were first formulated in 2016 (Rospocher et al., 2016) and have already been implemented in diverse domains, such as article processing (Gottschalk and Demidova, 2018a), news (Rospocher et al., 2016) or even tourism (Wu et al., 2020). In these cases, an Event-Centric Knowledge Graph (ECKG) is "*a Knowledge Graph in which all information is related to events through which the knowledge in the graph obtains a temporal dimension*" (Rospocher et al., 2016). Differently to regular knowledge graphs, where the information usually gravitates around a number of central entities, ECKGs put the focus on specific events, retrieving information about them from different sources and combining it in order to properly describe them.

Differently from this approach, this thesis aims to describe legal decisions using the events as the basis, being *blocks* that describe the legal judgment. A case is considered to be a narrative of events in different dimensions, namely *procedural* or *relative to the case under judgment*, and a case represented as a succession of events can be extremely useful for various applications within the legal domain. Since this concept is slightly different to the previous definition of what an ECKG is, the presented approach was named Event-Based Knowledge Graph, although both approaches share several common points, and the tools presented could also be used to build an ECKG.

In the legal domain, several proposals have recently delved into building knowledge graphs (Filtz, 2017), including initiatives such as the Lynx Project (Rodríguez-Doncel and Montiel-Ponsoda, 2020), that aims to build a multilingual knowledge graph to support compliance-related services. In spite of these recent efforts, none of them tackles event processing.

Unlike previous related works, such as EventKG (Gottschalk and Demidova, 2018a), there are no previous structured knowledge bases in the domain in order to help build the event-based knowledge graph, but only repositories with legal documents without annotations. Therefore, the first step had to be the retrieval and the processing of raw documents in order to extract relevant events from them. Although this approach focuses on events, as Event-Centric Knowledge Graphs do, events are not understood in the same way projects as EventKG did. The approach presented here processes and represents the relevant events (actions or happenings) mentioned in legal texts that shape the legal case, not events in the sense of a ceremony, a "named event" (like a specific war or regular sporting events such as the Olympic Games) or a journalistic event, with contributions from different sources. Even though eventually other types of resources could be integrated, such as news related to a case, or appeals to other courts such as nationals, the focus is kept on the events mentioned in a judgment. The definition of an Event-Based Knowledge Graph would be, therefore "a Knowledge Graph where information is represented as a series of events", although additional information can be introduced, such as the annotations from which the events were derived.

Once the events from the documents are extracted using the WhenTheFact event extractor from Section 8.2, they can be translated to RDF format, using the ontology (Section 9.1) and the converter (Section 9.2) expressly created for this purpose. Finally, the document annotations with the events extracted are sent to the knowledge graph, which can be later queried. Taking into consideration that the legal domain practitioners are not usually familiar with semantic web technologies, a service with a series of predefined queries is provided in order to facilitate consulting the knowledge graph.

The junction of the different resources and tools detailed in previous sections allow therefore to create a legal event-based knowledge graph. Fig. 9.10 shows how the different contributions interact in order to populate and query the knowledge graph.

First, the event extractor WhenTheFact process and annotates legal documents from two different European sources. Then, the annotated version of the document (in the EventsMatter format) is sent to the fromTimeToTime converter in order to be outputted as RDF, using the fromTimeToTime ontology. Afterwards, this result is updated to the knowledge graph, which is therefore populated with documents in

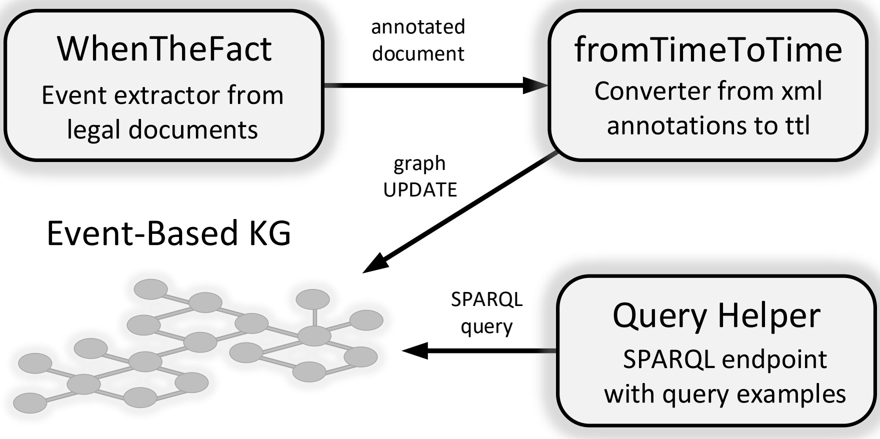


Figure 9.10: Pipeline of population and query of the legal event-based knowledge graph

the format ft3+events (an example was shown in Fig. 9.7). Finally, the graph can be queried from the SPARQL endpoint enabled for this purpose. In this endpoint, some basic predefined queries help to explore the knowledge graph (such as "return events form a specific year, document or type"), but also free queries can be sent to it.

Currently, the only way to add documents to the knowledge graph is via the WhenTheFact event extractor due to security reasons. Nevertheless, all the code and resources needed to replicate and handle the legal event-based knowledge graph are provided. It is also possible to choose the way to store the triples; for the tests, both Virtuoso[159] and BlazeGraph[160] have been used, and just the parameters of the request (such as the URL and the authentication, if needed) need to be adapted.

One of the main applications to exploit the knowledge graph is timeline generation, a task that has already been tackled for Event-Centric Knowledge Graphs in EventKG+TL (Gottschalk and Demidova, 2018b). Being able to build the timeline of the different actors involved in a case would also help to find inconsistencies in the alabi provided by them and other pieces of evidence. Additionally, the performance of general tasks such as Question Answering, already targeted in traditional Knowledge Graphs such as the one by the Lynx project (Rodríguez-Doncel and Montiel-Ponsoda, 2020), could be improved for the time-related questions, that could be much more precise and complex. Summarization tasks can also benefit from an event-based representation,

[159] `https://github.com/openlink/virtuoso-opensource`
[160] `https://github.com/blazegraph/database/releases/tag/BLAZEGRAPH_2_1_6_RC`

since event-based summarization techniques have already been explored in literature (Filatova and Hatzivassiloglou, 2004; Marujo et al., 2017). Moreover, reasoning systems and search engines can make use of event arguments in order to improve their results, being possible to refine event-based searches such as “Give me cases about car accidents where the driver was a man” or “Cases where the accident happened after a criminal action”.

Finally, one of the most interesting applications for law firms would be pattern recognition. The possibility of looking for previous judgments with similar narratives in terms of events and temporal spans would be an extremely valuable tool for legal practitioners, since it would really enhance the search of jurisprudence and would help to plan possible time-lapses in the resolution of the legal procedure.

Summary

This last chapter presented a series of tools that allow to create a Legal Event-Based Knowledge Graph. The approach is based on the assumption that the relevant events extracted from a legal judgment describe it in a way powerful to be exploited.

Once the annotation of a legal decision is done (using for instance the WhenTheFact event extractor), it is sent to the fromTimeToTime converter, a tool able to output a document in different annotation formats and as RDF. The tool converts the xml annotated document into a turtle file that includes both information about the document and its annotations and a special representation of all the events detected in the document, based on an ontology created for this purpose. Finally, the output is used to populate an Event-Based Knowledge Graph, that can be later queried from a SPARQL endpoint with some predefined queries to facilitate the task to people foreign to the Semantic Web. All the resources are freely available and can be combined with other tools in order to replicate or improve the functionality. In this chapter, the ft3 ontology and its usage together with WhenTheFact within the pipeline, that begins with the id of a document in an official European repository and ends with the storage of the information extracted in a knowledge graph that can be easily queried (including some example queries for people not familiar with linked data technologies) from the website of the tools, confirm hypotheses H2.b and H3.

Part IV

CONCLUSIONS

Chapter 10

Conclusions and Future Work

The main objective of this thesis is to improve the temporal information extraction and representation in the legal domain. The way to do so was analyzing the state of the art, the particularities of the legal domain and the lacks of temporal taggers, and then creating dedicated tools and new semantic resources that facilitate their visualization and use in further semantic tasks. This aim was tackled on the basis of a series of hypotheses; the main contributions resulting and their relation to the objectives explained in Chapter 4 are described in Section 10.1. Finally, the future work intended, some of them aiming at covering detected open problems not covered by this thesis, is stated in Section 10.3.

10.1 Contributions

In order to attain the main objective of the thesis, several sub-objectives (presented in Chapter 4) were derived in order to divide the tasks to address. These tasks derived from sub-objectives, at the same time, define the main research lines and steps in the thesis. They are outlined below:

O1.1. Identification of particularities and needs in the legal domain with regard to temporal information: this work is described in Chapter 3.

O1.2. Develop the resources needed to alleviate the particularities detected: different resources have been presented for temporal expressions (Chapter 5) and events (Chapters 7 and 9).

O2.1. Improve temporal information processing for the Spanish language: this objective was targeted with Añotador (Section 6.1).

O2.2. Improve temporal information processing for legal texts: this objective was targeted with the legal implementation of Añotador (Section 6.1) and the additional software lawORdate (Section 6.2).

O2.3. Improve representation of temporal information annotation: the different resources provided are described in Chapter 9.

O3.1. To be able to extract events from judgments: this objective is covered by the WhenTheFact software (Section 8.2).

O3.2. To be able to represent legal judgments as a series of events: the resources created to this aim are presented in Chapter 9.

O3.3. To use the judgment event-centric representation to build visual and intuitive ways to navigate through legal texts (such as timelines): this objective was tackled by the WhenTheFact timeline visualization (Section 8.2).

Regarding the hypothesis, the results of the tools provided show that tackling the particularities of the legal domain improve the performance with respect to using generic tools. Additionally, the creation of a knowledge graph based on events using the ontology presented evidence of the further semantic use of them by querying the graph.

10.2 Ethical and Legal Compliance

This thesis is aligned with several principles of the Asilomar Artificial Intelligence Principles[161], since it aims to make legal documents accessible to people.

10.3 Future Work

Regarding future work, as described in Section 6.3, there are still several open problems in the processing of temporal information. Besides targeting the context-dependent temporal expressions described there, future work envisaged is outlined below.

[161] https://futureoflife.org/ai-principles/

Extending the event extraction to more languages One of the ideas I had no time to implement for the event extraction tool WhenTheFact was to extend the annotation to other languages. Since the ECJ legal decisions are usually available in all the languages of the European Union, it would be a nice implementation to annotate in English but allow the user to navigate the document in another language. Nevertheless, the current problem is to find the correct extent of the correspondence between annotations in English and the new language, since for instance languages such as German have different sentence structures. Although several approaches have been tested already, none of them has been good enough to guarantee acceptable results for all the languages.

Extending the corpora available Despite the extensive work on corpora creation done during this thesis, there is still a lot of work to be done in the domain. Similarly to the idea previously presented, one possible idea to semi-automatically increase the existing corpora is to annotate multilingual datasets in one language and then automate the annotation for the other languages. However, the bottleneck remains the need for native specialists to ensure correct annotation.

Event co-reference Besides improving event extraction with regard to the types of events, also co-reference should be implemented. Currently, every mention of an event is considered to be independent, but this is not necessarily true, and covering this would allow representing judgments in a much more faithful way.

Enriching the knowledge graph with metadata Mainly metadata not related to the temporal information, such as the actors involved in the cases. This for instance would help to solve coreference, since currently just the textual mention is got, that can consist of pronouns. Once this is achieved, queries will be able to retrieve for instance the timeline of one actor's involvement in a case.

Processing more types of documents I am currently specialized in processing legal decisions, but there are many other legal documents I could target, such as contracts. Additionally, in the Spanish jurisdiction, it would be useful to process documents such as the BOE (Spanish national gazette).

Deep Learning for covering more events Although legal events tend to be repeated in all the decisions of the same court, one of the main problems encountered was to identify the events related to the facts under judgments, since they depend very much on the case itself. In this regard, one possible way to improve identification would be to massively train a deep learning system and relate the relevant events to the articles cited in each judgment.

Facilitate the queries to the EBKG Since one of the target users of the contributions presented are legal practitioners, usually foreign to SPARQL, one of the planned improvements is to adapt Natural Language queries to SPARQL translators to the legal domain terminology. This would help to boost the use of these technologies, as well as to bring the Semantic Web technologies and the legal domain closer together.

Further exploit the knowledge graph Finally, one of the envisaged tasks building on the work already done, is to exploit the legal event-based knowledge graph with event-driven applications, such as event-based summarization, event-driven search or a question answering system.

Part V

APPENDIX

ANNEX A

Questionnaire

The following document was distributed in July 2019 among Lynx partners in order to detect the needs of time expression extraction and normalization in legal documents and any temporal expressions not considered by the TimeML standard.

QUESTIONNAIRE

Introduction on temporal expressions:

Temporal Expressions are any word or sequence of words referring to a time instant (e.g. ‘five o’clock’) or a time interval (e.g. ‘from nine to ten’). Temporal expressions frame events or happenings implicitly or explicitly mentioned in the document. Following the ISO-TimeML standard, we distinguish among dates, times, durations and sets; additionally, we also plan to add intervals.

- DATE: Calendar expressions such as 'October 7, 1991', '22/01/2018', or '1992'; also relative expressions like 'Two days ago'.
- TIME: Points in time ('At seven o'clock', '22:30', '3.30pm'...), absolute or relative ('Half an hour ago', 'In two minutes and three seconds').
- DURATION: Amounts of time like 'Two days', 'Three years and six months', 'Two centuries', 'One hour and 20 minutes' or 'Half an hour'.
- SET: Repetitions in time (such as 'Monthly', 'Twice a week', 'Every Monday', 'Three times a year', 'Every first of the month'...).
- INTERVAL: Period between two temporal expressions (‘from 14h to 20h’, ‘from Monday to Friday’…).

In general, we can consider a temporal expression is any expression than answers “when” and “how long”.

Nevertheless, current temporal taggers (the tools that identify and normalize, this is, give a standard value to these expressions) are not prepared to deal with legal documents. They are not able, for instance, to detect expressions such as “five working days”, since they are not usually found in other domains.

To be able to properly improve the Lynx service to detect these expressions, we would like you to answer the following questions about your needs and interests regarding temporal expressions.

1. Temporal expressions of interest.

Is there any specific temporal expression you would like to find that is not common to other domains? (such as “working days”). Please don’t hesitate to contact us for doubts. Anything you would like to find/has any time-relate value to you is of interest.

Temporal expression	Different ways to express it (languages, different ways to express it)	Example/Other comments
five working days	días laborables (es), X working days	...
years of contribution	worked years (en), years of contribution (en), años cotizados (es)	...
the date of signature	...	...
...	...	...

2. Temporal expressions to refer.

Usually temporal expressions are relative, such as if for instance we say “two days ago”. Temporal taggers tend to normalize with regard to the present day or to the date of creation of the document. Is there any other type of date you would be interested to use as anchor date, such as the date of publication of a document or the date of signature? It can vary depending on the type of document. Please let us know any relevant metadata related to time you would like to consider to this aim, or other considerations we are not taking into account. If this information is not available as metadata, any hints on where to find it within the text is also helpful.

Temporal expression	Type of document/other info	Where to find this information in the text if not available as metadata	Other comments
Date of signature	Contracts	End of the document	...
Date of the decision	worked years (en), years of contribution (en), años cotizados (es)	End of the document	...
...	...	...	...

3. Useful temporal expressions

Maybe not al the temporal expressions in a text are necessarily useful. For instance, some legal references, such as “European Council of the 3 June 2018” or “Real Decreto-ley 10/2018, de 24 de agosto” can include them without being actually a date in the text, but part of a legal reference. Do you have any preference on how to deal with it?

☐ I would mark them all.
☐ I would mark them all, but distinguishing them somehow (such as having different types or timelines in the text).
☐ I would just mark expressions from the text, not the ones that are part of a legal reference.
☐ No strong opinion, not important.

4. Other comments

Please let us know any other comments on how to deal with temporal expressions in the Lynx service:

ANNEX B

Flow Chart of Añotador

The steps of the flow chart of Añotador (Fig. B.1) are outlined below:

A) Collection of the parameters: the text to annotate (compulsory), the anchor date (optional, if not provided the current date will be used), output format (if not provided, by default TimeML format will be used).

B) We use CoreNLP to preprocess the text. In the case of Spanish, we use IxaPipes models.

C) The rules developed for the identification of the temporal expressions are applied following a certain order, sometimes in an iterative way. We first detect the basic tokens that might be relevant for the temporal information (such as weekdays, numbers or granularities) and we give them a value (e.g., "Mil novecientos noventa y nueve", "Nineteen ninety-nine", is valued as "1999"). Once these tokens have been identified and evaluated, we look for basic temporal expressions comprising them (such as "two days", that is a "number + granularity"). Finally, we look for compositions of these basic expressions. This processing returns the following information for each of the identified temporal expressions:

   (a) Type: the type of the temporal expression from among DATE (calendar expressions), DURATION, TIME (clock expressions) or SET (an expression that repeats periodically over time).

   (b) Value: the value given to the expression. It is not necessarily the final value, they can be functions predefined in the system, such as *anchor(A,B,C)*, where

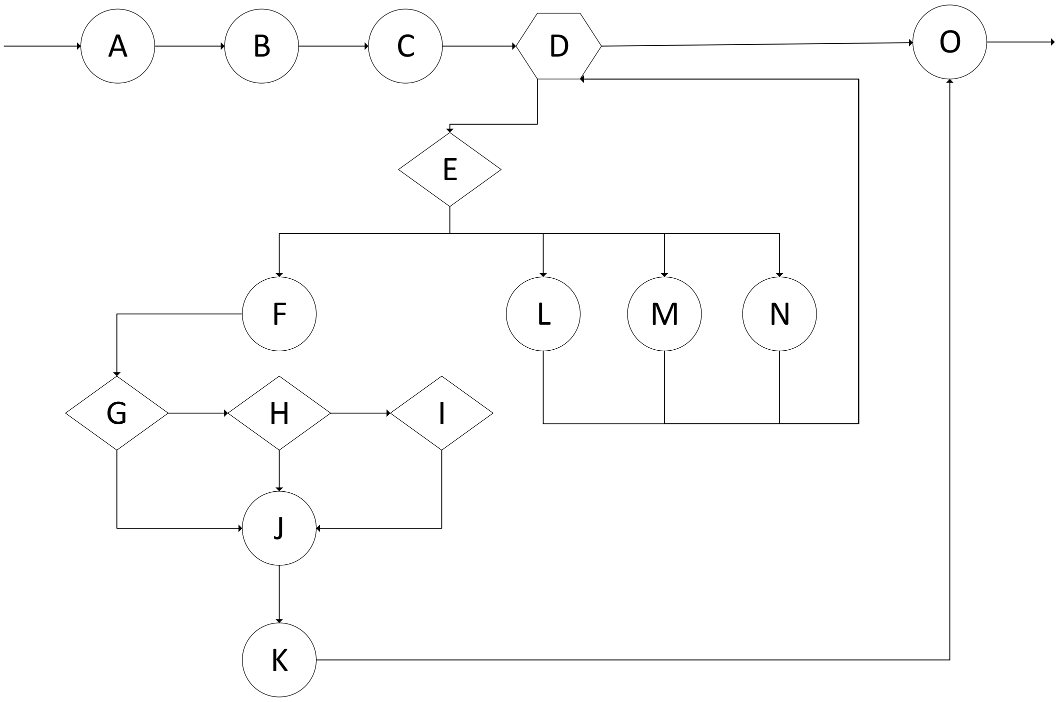


Figure B.1: Flow Chart of the software Añotador.

A is the anchor date (it can be a variable), B the operation to be carried out and C a duration. In the case for instance of *anchor(TODAY,-,1D)*, it implies that the value of the temporal expression will subtract one day to the anchor date (this would be for instance the return of the expression "yesterday").

(c) Freq: just used in the expression of type SET. It represents the amount of times the temporal expression is repeated in the period expressed in value. For instance, "Twice a week" would have as value "P1W" and "2D" as freq.

(d) Mod: This is an optional value for modifiers such as BEFORE, START, MID or APPROX.

(e) Rule: the name of the last rule applied; it is used for tracing the internal processing of the rules and does not affect the result.

An example of the output would be the following:

("DATE", "anchor(TODAY,+,2D)", "", "", "Rule$PasadoMañana")

D) From the output of the rules, we iterate over each sentence of the text, checking each temporal expression and processing it using a normalization algorithm that comprises the following steps:

E) We first check the type of the temporal expression, since the four types require different normalizations.

  F) If the output of the rules says that the expression is of the type DATE, we first normalize the unknown values (expressed as X in the output, meaning that for instance "XXXX-01-23" would mean that we detected the expression "23rd January", but without an explicit mention of the year), using as reference the date provided to this aim (anchor date) by the user, the current date, or another date provided in the same sentence.

    G) We then check if it is a reference like "the last summer", and if so we perform the pertaining calculation. In the example case, we would check if the mentioned summer refers to the present year (if the reference date was December, for instance), or to the previous year (if the reference date was before the current year summer, in March, for instance).

    H) If it is not that kind of reference, we check if it is anchored to the current date, such as "the day before yesterday", whose value according to the rules would be "anchor(TODAY,-,2D)". If this was the case, we would do the appropriate calculations.

    I) If the previous condition is neither fulfilled, we check other types of anchoring implemented, such as for instance "this month" or "this semester", where we must use our reference date but using a different granularity than a full date. If our reference date was for instance "this quarter", if our reference date was for instance "2020-02-11", the final value would be "2020-Q1", since this date belongs to the first quarter of 2020.

    J) Depending on the type of value we obtained from the rule, we will do the required transformation to obtain an expression normalized according to the ISO-8601 standard (or to our additions to it), as explained in each of the examples of the steps G, H and I.

K) Once obtained the final value of the temporal expression, we store the value of the dates to use as reference in future normalizations within the same sentence, and we proceed to the processing of the next temporal expression, starting again the iteration (node D).

L) If it is an expression of the type TIME, we add the reference date to anchor it to the calendar.

M) If the type is DURATION, we process it to adapt it to the TimeML standard.

N) If the type is SET, no processing is needed.

O) Once we have processed all the temporal expressions in the text, in the case of the legal implementation we look for intervals. Finally, we transform the result into the requested format (NIF, JSON or TimeML) and we return it annotated.

ANNEX C

CENDOJ certificate

El Centro de Documentación Judicial CENDOJ concede autorización a la solicitante, **DOÑA MARÍA NAVAS LORO**, titular del DNI número 53560559E, Estudiante de Doctorado en el Grupo de Ingeniería de Ontología, Departamento de Inteligencia Artificial, Escuela Técnica Superior de Ingenieros Informáticos (Universidad Politécnica de Madrid), para hacer uso de las resoluciones contenidas en la base de Datos de Jurisprudencia del Consejo General del Poder Judicial que se expresarán.

El objetivo, de acuerdo con lo manifestado en la solicitud, ha sido proceder a la consulta y análisis de aquellas resoluciones que resulten de su interés para la elaboración de su **tesis doctoral cuyo fin es desarrollar un framework capaz de tratar expresiones temporales en documentos jurídicos** y obtener recursos de PNL dedicados en el dominio temporal y legal (como conjuntos de reglas y marcos semánticos) capaz de detectar datos relevantes temporales en documentos sin procesar; ampliación temporal de las opciones de representación existentes y obtención de un etiquetador temporal enfocado en el dominio legal preparado para tener en cuenta información contextual.

Las resoluciones han sido seleccionadas por el CENDOJ y entregadas a la reutilizadora, anotados, para que a partir de ellos, pueda desarrollar el objeto de su tesis.

En cuanto al uso que no constituya mera consulta, sino reutilización regulada por la Ley 37/2007 sobre reutilización de la información del sector público, dichas resoluciones podrán obtenerse asimismo a través del buscador público reseñado con los requisitos que se expresarán a continuación.

La presente autorización no se encuentra sujeta a contraprestación económica alguna pero sí condicionada al cumplimiento de los siguientes requisitos específicos:

1.- Esta utilización no podrá tener, en ningún caso, una finalidad comercial.

2.- El contenido de la información obrante en la resolución judicial, incluyendo sus metadatos, no podrá ser objeto de modificaciones o alteraciones de ningún tipo, ni podrá desnaturalizarse el sentido de la información.

3.- Ha de citarse la fuente de origen, CENDOJ.

4.- Queda prohibida la reversión del procedimiento de disociación de datos de carácter personal mediante la adición de nuevos datos obtenidos de otras fuentes o de otro modo.

5.- En el supuesto de publicar el texto original traducido a otro idioma, habrá de mencionarse expresamente dicha circunstancia, indicando que el idioma original de la fuente es el español o la lengua cooficial de que se trate.

6.- La reutilizadora, adquiere el compromiso de guardar absoluta confidencialidad de cualesquiera datos de carácter personal a los que pudiera haber tener acceso a través de los actos de reutilización que se describen, antes, durante y al término de la conclusión de la tesis.

DIRECTOR DEL CENTRO DE DOCUMENTACIÓN JUDICIAL

ANNEX D

fromTimetoTime ORSD

This annex presents the ontology requirement specification document (ORSD) of the fromTimeToTimeOntology, drafted in accordance with the LOT methodology (Poveda-Villalón et al., 2019). More information about this ontology can be found in Section 9.1 of this thesis.

From Time To Time Ontology (ft3:)
1. Purpose
The purpose of this ontology is to represent in a semantic fashion the temporal information present in text, harmonizing existing non-ontological standards and adding relevant information not included in those.
2. Scope
Representation of temporal information annotation and events. Facilitate the event-based representation of a document, stressing the temporal information annotation information and allowing different formats of temporal information.
3. Implementation Language
OWL
4. Intended End-Users
User 1. People involved in temporal tagging. User 2. People needing representation of temporal information. User 3. NLP practicioners in general. User 4. LinkedData users and KnowledgeGraph builders.
5. Intended Uses
Use 1. Event-based representation of information. Use 2. Facilitate translation among annotation formats and ontologies. Use 3. Storage of annotations for latter tasks (e.g. visualization, search). Use 4. Representation of events for different uses (e.g. timeline generation, pattern recognition).

<table>
<tr><th colspan="4">From Time To Time Ontology (ft3) Specification Document</th></tr>
<tr><th colspan="4">6. Ontology Requirements</th></tr>
<tr><th colspan="4">a. Non-Functional Requirements</th></tr>
<tr><td colspan="4">NFR 1. It must be able to represent the different arguments of TimeML annotations.
NFR 2. It must facilitate the transition between one annotation standard to another, or at least establish the relationship.
NFR 3. It must be able to represent fuzziness in time.
NFR 4. It must be able to represent the different levels of abstraction of an event.
NFR 5. It must allow to manage opposing events.
NFR 6. It must allow to have "composed" temporal expressions (e.g. "It happened on Monday or Tuesday.", "Every other day.").
NFR 7. It must allow SET representation.</td></tr>
<tr><th colspan="4">b. Lists or tables of requirements written as Competency Questions and sentences</th></tr>
<tr><td colspan="2">CQG1. XXXX</td><td colspan="2">CQG2. YYYY</td></tr>
<tr><td colspan="2">CQ1. What are the elements of an annotation?
CQ2. What are the correspondences between different annotation formats?
CQ3. What are the main abstract standard concepts described by temporal annotations?</td><td colspan="2">CQ4. What are the correspondences between temporal annotations and temporal information in other ontologies?
CQ5. How should different concepts of events be represented?</td></tr>
<tr><th colspan="4">7. Pre-Glossary of Terms</th></tr>
<tr><th colspan="4">a. Terms from Competency Questions</th></tr>
<tr><td>Annotation
Annotation Format</td><td>Abstract Concept
Temporal Information</td><td>Correspondence
Event</td><td>Standard</td></tr>
<tr><th colspan="4">b. Terms from Answers</th></tr>
<tr><td>Annotation
Argument
Granularity
Event annotation</td><td>Annotation Standard
Type/Class
Value
Temporal annotation</td><td>Time ontology
Event schematization
Event instance
SET</td><td>When
Who
What
Where</td></tr>
<tr><th colspan="4">c. Objects</th></tr>
<tr><td colspan="4">value, type, class, id, Event_who, Event_when, Event_what...</td></tr>
</table>

ANNEX E

Glossary

Glossary of acronyms in the document.

- BOE: Boletín Oficial del Estado, the Spanish Official State Gazette.
- CDTE: Context-dependent Temporal Expression.
- CENDOJ: Centro de Documentación Judicial (National Center of Judicial Documentation).
- CRF: Conditional Random Fields, a machine learning technique.
- DCT: Document Creation Time. It is frequently used as an anchor date for normalization.
- EBKG: Event-Based Knowledge Graph.
- ECHR: European Court of Human Rights, source of part of the documents in the TempCourt corpus and the EventsMatter corpus.
- ECKG: Event-Centric Knowledge Graph.
- ECJ: European Court of Justice, source of part of the documents in the TempCourt corpus.
- GATE: General Architecture for Text Engineering.
- NER: Named Entity Recognizer.

- NLP: Natural Language Processing.
- POS: Part of Speech.
- QA: Question Answering, an NLP task.
- SVM: Support Vector Machines, a machine learning technique.
- TE: Temporal Expression, also called Time Expression.
- UIMA: Unstructured Information Management applications from Apache.
- USSC: United States Supreme Court, source of part of the documents in the TempCourt corpus.

Bibliography

ACE English Annotation Guidelines for Events (2005). ACE (Automatic Content Extraction) English Annotation Guidelines for Events . `https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf`. 21

ACE2005 (2005). The ACE 2005 (ACE 05) Evaluation Plan. Evaluation of the Detection and Recognition of ACE Entities, Values, Temporal Expressions, Relations, and Events. `https://www.semanticscholar.org/paper/The-ACE-2005-(-ACE-05-)-Evaluation-Plan-Evaluation-NTRODUCTION/3a9b136ca1ab91592df36f148ef16095f74d009e`. 48, 52

Agerri, R., Bermudez, J., and Rigau, G. (2014). IXA pipeline: Efficient and Ready to Use Multilingual NLP tools. In *LREC*, volume 2014, pages 3823–3828. 47, 110

Aguilar, J., Beller, C., McNamee, P., Van Durme, B., Strassel, S., Song, Z., and Ellis, J. (2014). A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 45–53. 20

Ahn, D. (2006). The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8. Association for Computational Linguistics. 20

Alam, M., Gangemi, A., Presutti, V., and Recupero, D. R. (2021). Semantic role labeling for knowledge graph extraction from text. *Progress in Artificial Intelligence*, pages 1–12. 47

Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., and Lampos, V. (2016). Predicting judicial decisions of the european court of human rights: A natural language processing perspective. *PeerJ Computer Science*, 2:e93. 66

Allen, J. F. and Ferguson, G. (1994). Actions and Events in Interval Temporal Logic. *Journal of Logic and Computation*, 4(5):531–579. 26

Angeli, G., Manning, C. D., and Jurafsky, D. (2012). Parsing time: Learning to interpret time expressions. In *Proceedings of NAACL 2012: Human Language Technologies*, pages 446–455. 19

Apache Software Foundation (2014). OpenNLP Natural Language Processing Library. http://opennlp.apache.org/. 44

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Vol 1*, ACL '98/COLING '98, page 86–90. Association for Computational Linguistics. 32, 169

Bentivogli, L., Forner, P., Magnini, B., and Pianta, E. (2004). Revising the WordNet domains hierarchy: semantics, coverage and balancing. In *Proceedings of the workshop on multilingual linguistic resources*, pages 94–101. 33

Bertoldi, A., Chishman, R., Rigo, S. J., and Minghelli, T. D. (2014). Cognitive linguistic representation of legal events. towards a semantic-based legal information retrieval. In *COGNITIVE 2014 : The Sixth International Conference on Advanced Cognitive Technologies and Applications*. 49

Bethard, S. (2013). ClearTK-TimeML: A minimalist approach to TempEval 2013. In *Proceedings of the Workshop SemEval 2013*, pages 10–14. ACL. 39, 40, 93

Bies, A., Song, Z., Getman, J., Ellis, J., Mott, J., Strassel, S., Palmer, M., Mitamura, T., Freedman, M., Ji, H., et al. (2016). A comparison of event representations in DEFT. In *Proceedings of the Fourth Workshop on Events*, pages 27–36. 21

Boroditsky, L. (2011). How languages construct time. In *Space, time and number in the brain*, pages 333–341. Elsevier. 3

Butt, M. (2010). *The light verb jungle: still hacking away*, page 48–78. Cambridge University Press. 168

Capet, P., Delavallade, T., Nakamura, T., Sandor, A., Tarsitano, C., and Voyatzi, S. (2008). A risk assessment system with automatic extraction of event types. In *International Conference on Intelligent Information Processing*, pages 220–229. 46

Cervantes, I. (2019). El espaÑol: Una lengua viva. informe 2019. Technical report, Instituto Cervantes. 99

Chambers, N. (2013). Navytime: Event and time ordering from raw text. In *Proceedings of the Workshop SemEval 2013*, volume 2, pages 73–77. 40

Chambers, N., Cassidy, T., McDowell, B., and Bethard, S. (2014). Dense event ordering with a multi-pass architecture. *Transactions of ACL*, 2:273–284. 39, 93

Chambers, N. and Jurafsky, D. (2008). Unsupervised learning of narrative event chains. *Proceedings of ACL-08: HLT*, pages 789–797. 46

Chambers, N. and Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on NLP of the AFNLP: Vol 2*, pages 602–610. Association for Computational Linguistics. 46

Chang, A. X. and Manning, C. D. (2012). SUTIME: A Library for Recognizing and Normalizing Time Expressions. In *LREC*, volume 2012, pages 3735–3740. 37, 38, 41, 93, 132

Chang, A. X. and Manning, C. D. (2014). TokensRegex: Defining cascaded regular expressions over tokens. Technical Report CSTR 2014-02, Department of Computer Science, Stanford University. 37, 110, 160

Chasin, R. (2010). Event and temporal information extraction towards timelines of wikipedia articles. *Simile*, pages 1–9. 53

Chen, H. W., Du, J., Song, H.-Y., Liu, X., Jiang, G., and Tao, C. (2018). Representation of time-relevant common data elements in the cancer data standards repository: statistical evaluation of an ontological approach. *JMIR medical informatics*, 6(1):e7. 182

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46. 92

Comisión de expertos Modernización del lenguaje jurídico, M. d. J. (2011). Informe de la Comisión de Modernización del Lenguaje Jurídico. 5

Cunningham, H., Tablan, V., Roberts, A., and Bontcheva, K. (2013). Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. *PLOS Computational Biology*, 9(2):1–16. 36, 90, 125, 150, 163

Danilova, V. (2015). *Linguistic support for protest event data collection.* Universitat Autònoma de Barcelona,. 23

Das, D., Chen, D., Martins, A. F., Schneider, N., and Smith, N. A. (2014). Frame-semantic parsing. *Computational linguistics*, 40(1):9–56. 47

Dell'Orletta, F., Marchi, S., Montemagni, S., Plank, B., and Venturi, G. (2012). The SPLeT–2012 Shared Task on Dependency Parsing of Legal Texts. In *Semantic Processing of Legal Texts (SPLeT-2012) Workshop Programme*, page 42. 55, 160

Derczynski, L. and Gaizauskas, R. (2010). USFD2: Annotating temporal expresions and TLINKs for TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 337–340. ACL. 42, 93

Dias, G. H., Hasanuzzaman, M., Ferrari, S., and Mathet, Y. (2014). Tempowordnet for sentence time tagging. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 833–838. 31

Ermolayev, V., Batsakis, S., Keberle, N., Tatarintseva, O., and Antoniou, G. (2014). Ontologies of time: Review and trends. *International Journal of Computer Science & Applications*, 11(3). 20

Etiene, T., Pagliosa, P., and Nonato, L. G. (2015). Linea: Building Timelines from Unstructured Text. In *28th SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2015*, pages 234–241. IEEE Computer Society. 46

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874. 39

Fellbaum, C. (1998). *WordNet: An electronic lexical database (Language, Speech and Communication)*. MIT Press. 38

Feng, X., Qin, B., and Liu, T. (2018). A language-independent neural network for event detection. *Science China Information Sciences*, 61(9):1–12. 46

Fernández-López, M. and Gómez-Pérez, A. (2004). Searching for a time ontology for semantic web applications. In *Proceedings of the Formal Ontology in Information Systems. Third Interntacional Conference (FOIS-2004)*. IOS Press. 20

Ferro, L., Gerber, L., Mani, I., Sundheim, B., and Wilson, G. (2005). TIDES 2005 standard for the annotation of temporal expressions. Technical report, MITRE. 16

Ferro, L., Mani, I., Sundheim, B., and Wilson, G. (2001). TIDES temporal annotation guidelines-version 1.0.2. Technical report, MITRE. 19, 30

Filannino, M. and Nenadic, G. (2015). Temporal expression extraction with extensive feature type selection and a posteriori label adjustment. *Data & Knowledge Engineering*, 100:19 – 33. 43

Filatova, E. and Hatzivassiloglou, V. (2004). Event-based extractive summarization. In *Text Summarization Branches Out*, pages 104–111. 197

Filtz, E. (2017). Building and processing a knowledge-graph for legal data. In *European Semantic Web Conference*, pages 184–194. Springer. 194

Filtz, E., Navas-Loro, M., Santos, C., Polleres, A., and Kirrane, S. (2020). Events matter: Extraction of events from court decisions. In *Proceedings of the 33rd International Conference on Legal Knowledge and Information Systems (JURIX 2020).*, pages 33–42. 13, 148, 164, 176, 177

Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612. 42

Fulda, J., Brehmer, M., and Munzner, T. (2015). TimeLineCurator: Interactive Authoring of Visual Timelines from Unstructured Text. *IEEE transactions on visualization and computer graphics*, 22(1):300–309. 46

Gandon, F., Governatori, G., and Villata, S. (2017). Normative requirements as linked data. In *The 30th international conference on Legal Knowledge and Information Systems (JURIX 2017).* 29

Gangemi, A., Presutti, V., Recupero, D. R., Nuzzolese, A. G., Draicchio, F., and Mongiovì, M. (2017). Semantic Web Machine Reading with FRED. *Semantic Web*, 8(6):873–893. 12, 47

Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N. F., Peters, M., Schmitz, M., and Zettlemoyer, L. S. (2017). AllenNLP: A Deep Semantic Natural Language Processing Platform. In *arXiv preprint arXiv:1803.07640.* 45, 47

Garijo, D. (2017). WIDOCO: a wizard for documenting ontologies. In *International Semantic Web Conference*, pages 94–102. Springer, Cham. 181, 187

Gildea, D. and Jurafsky, D. (2002). Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3):245–288. 38

Gordon, T. F. (2008). The legal knowledge interchange format (LKIF). *Estrella deliverable d4*, 1. 29

Gottschalk, S. and Demidova, E. (2018a). EventKG: A Multilingual Event-Centric Temporal Knowledge Graph. In *European Semantic Web Conference*, pages 272–287. Springer. 24, 194, 195

Gottschalk, S. and Demidova, E. (2018b). EventKG+TL: Creating Cross-Lingual Timelines from an Event-Centric Knowledge Graph. In *European Semantic Web Conference*, pages 164–169. Springer. 196

Guda, V., Srujana, I., and Naik, V. (2011). Reasoning in Legal Text Documents with Extracted Event Information. *International Journal of Computer Applications*, 28:8—13. 44, 50, 85

Hagege, C. and Tannier, X. (2008). XTM: A robust temporal text processor. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 231–240. Springer. 45

Han, B., Gates, D., and Levin, L. (2006). From language to time: A temporal expression anchorer. In *Thirteenth International Symposium on Temporal Representation and Reasoning (TIME'06)*, pages 196–203. IEEE. 43

Hearst, M., Dumais, S., Osuna, E., Platt, J., and Scholkopf, B. (1998). Support Vector Machines. *IEEE Intelligent Systems and their Applications*, 13(4):18–28. 39

Hellmann, S. (2012). NIF: An ontology-based and linked-data-aware NLP Interchange Format. Technical report. 19

Hepple, M., Setzer, A., and Gaizauskas, R. (2007). USFD: preliminary exploration of features and classifiers for the TempEval-2007 tasks. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval 2007)*, pages 438–441. 42

Hoekstra, R., Breuker, J., Di Bello, M., and Boer, A. (2009). LKIF Core: Principled Ontology Development for the Legal Domain. In *Frontiers in Artificial Intelligence and Applications*, volume 188, pages 21–52. 29

Hogenboom, F., Frasincar, F., Kaymak, U., and De Jong, F. (2011). An overview of event extraction from text. In *Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011) at Tenth International Semantic Web Conference (ISWC 2011)*, volume 779, pages 48–57. CEUR-WS. 46

Honnibal, M., Montani, I., Van Landeghem, S., and Boyd, A. (2020). spaCy: Industrial-strength Natural Language Processing in Python. 44

Ingolfo, S., Jureta, I., Siena, A., Perini, A., and Susi, A. (2014). Nòmos 3: Legal compliance of roles and requirements. In *Conceptual Modeling*, pages 275–288, Cham. Springer International Publishing. 50

Isemann, D., Ahmad, K., Fernando, T., and Vogel, C. (2013). *Temporal Dependence in Legal Documents*, pages 497–504. Springer Berlin Heidelberg, Berlin, Heidelberg. 43, 85

Ji, H., Grishman, R., Chen, Z., and Gupta, P. (2009). Cross-document event extraction and tracking: Task, evaluation, techniques and challenges. In *Proceedings of the International Conference RANLP-2009*, pages 166–172. 51

Jung, H., Allen, J., Blaylock, N., de Beaumont, W., Galescu, L., and Swift, M. (2011). Building timelines from narrative clinical records: initial results based-on deep natural language understanding. In *Proceedings of BioNLP 2011 Workshop*, pages 146–154. ACL. 46

Kajiwara, T. and Komachi, M. (2016). Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings. In *Proceedings of COLING 2016: Technical Papers*, pages 1147–1158. 97

Kay, C., Roberts, J., Samuels, M., and Wotherspoon, I. (2009). *Historical thesaurus of the Oxford English dictionary.* Oxford University Press. 23

Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2006). Extending VerbNet with Novel Verb Classes. In *LREC*, pages 1027–1032. 32

Kitchenham, B. and Brereton, P. (2013). A systematic review of systematic review process research in software engineering. *Information and software technology*, 55(12):2049–2075. 13, 14

Kiyavitskaya, N., Zeni, N., Breaux, T. D., Antón, A. I., Cordy, J. R., Mich, L., and Mylopoulos, J. (2008). Automating the extraction of rights and obligations for regulatory compliance. In *International Conference on Conceptual Modeling*, pages 154–168. Springer. 49, 50

Kolya, A. K., Kundu, A., Gupta, R., Ekbal, A., and Bandyopadhyay, S. (2013). JU_-CSE: A CRF based approach to annotation of temporal expression, event and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 64–72. 43

Krisnadhi, A. and Hitzler, P. (2017). A core pattern for events. *Advances in Ontology Design and Patterns*, 32:29–38. 29

Lafferty, J. D., McCallum, A., and Pereira, F. C. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning (ICML 2001)*, pages 282–289. 38, 39

Lagos, N., Segond, F., Castellani, S., and O'Neill, J. (2010). Event extraction for legal case building and reasoning. In *International Conference on Intelligent Information Processing*, pages 92–101. Springer. 48, 50

Lee, K., Artzi, Y., Dodge, J., and Zettlemoyer, L. (2014). Context-dependent semantic parsing for time expressions. In *Proceedings of the 52nd Annual Meeting of the ACL*, volume 1, pages 1437–1447. 41, 42, 93, 140

Li, F., Du, J., He, Y., Song, H.-Y., Madkour, M., Rao, G., Xiang, Y., Luo, Y., Chen, H. W., Liu, S., et al. (2020). Time event ontology (TEO): to support semantic representation and reasoning of complex temporal relations of clinical events. *Journal of the American Medical Informatics Association*, 27(7):1046–1056. 26

Lippi, M., Lagioia, F., Contissa, G., Sartor, G., and Torroni, P. (2018). Claim Detection in Judgments of the EU Court of Justice. In *AI Approaches to the Complexity of Legal Systems*, pages 513–527, Cham. Springer International Publishing. 66

Llorens, H., Chambers, N., UzZaman, N., Mostafazadeh, N., Allen, J., and Pustejovsky, J. (2015). Semeval-2015 task 5: QA TEMPEVAL - Evaluating Temporal Information Understanding with Question Answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800. 52

Llorens, H., Derczynski, L., Gaizauskas, R. J., and Saquete, E. (2012). TIMEN: An Open Temporal Expression Normalisation Resource. In *LREC*, pages 3044–3051. 39

Llorens, H., Saquete, E., and Navarro, B. (2010). TIPSem (English and Spanish): Evaluating CRFs and Semantic Roles in TempEval-2. In *Proceedings of the Workshop SemEval*, pages 284–291. ACL. 38, 93

Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. In *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*. 43, 44, 172

Mani, I., Verhagen, M., Wellner, B., Lee, C. M., and Pustejovsky, J. (2006). Machine learning of temporal relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 753–760. Association for Computational Linguistics. 42

Mani, I. and Wilson, G. (2000). Robust temporal processing of news. In *Proceedings of the 38th annual meeting on ACL*, pages 69–76. 40

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the ACL, System Demonstrations*, pages 55–60. 37, 45, 160, 174

Marsh, E. and Perzanowski, D. (1998). MUC-7 evaluation of IE technology: Overview of results. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998.* 22

Marujo, L., Ribeiro, R., Gershman, A., de Matos, D. M., Neto, J. P., and Carbonell, J. (2017). Event-based summarization using a centrality-as-relevance model. *Knowledge and Information Systems*, 50(3):945–968. 197

Maxwell, K. T., Oberlander, J., and Lavrenko, V. (2009). Evaluation of semantic events for legal case retrieval. In *Proceedings of the WSDM'09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 39–41. ACM. 48, 50

Mazur, P. and Dale, R. (2009). The DANTE Temporal Expression Tagger. In *Human Language Technology. Challenges of the Information Society*, pages 245–257. 43

Mazur, P. and Dale, R. (2010). WikiWars: A New Corpus for Research on Temporal Expressions. In *Proceedings of EMNLP 2010*, pages 913–922. ACL. 30

McCallum, A. K. (2002). MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu. 39

Medvedeva, M., Vols, M., and Wieling, M. (2020). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, 28(2):237–266. 89

Mencía, E. L. and Fürnkranz, J. (2010). Efficient multilabel classification algorithms for large-scale problems in the legal domain. In *Semantic Processing of Legal Texts*, pages 192–215. Springer. 88

Minard, A.-L., Speranza, M., Urizar, R., Altuna, B., Van Erp, M., Schoen, A., and Van Son, C. (2016). MEANTIME, the NewsReader Multilingual Event and Time Corpus. In *Proceedings of LREC 2016*. 30

Minsky, M. (1975). *A Framework for Representing Knowledge. Reprinted in The Psychology of Computer Vision, P. Winston*. McGraw-Hill. 159

Mitamura, T., Yamakawa, Y., Holm, S., Song, Z., Bies, A., Kulick, S., and Strassel, S. (2015). Event nugget annotation: Processes and issues. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 66–76. 22

Navas-Loro, M. (2017). LawORDate: a Service for Distinguishing Legal References from Temporal Expressions. In *Proceedings of the 1st Workshop on Technologies for Regulatory Compliance co-located with the 30th International Conference on Legal Knowledge and Information Systems (JURIX 2017)*, pages 25–31. 13, 59, 61, 132

Navas-Loro, M., Filtz, E., Rodríguez-Doncel, V., Polleres, A., and Kirrane, S. (2019a). TempCourt: evaluation of temporal taggers on a new corpus of court decisions. *The Knowledge Engineering Review*, 34:e24. 12, 13, 36, 56, 57, 58, 61, 62, 87, 94, 97

Navas-Loro, M. and Rodríguez-Doncel, V. (2020). Annotador: a temporal tagger for Spanish. *Journal of Intelligent & Fuzzy Systems*, 39:1979–1991. 2. 13, 19, 99, 174

Navas-Loro, M. and Santos, C. (2018). Events in the legal domain: first impressions. In *Proceedings of the 2nd Workshop on Technologies for Regulatory Compliance co-located with the 31st International Conference on Legal Knowledge and Information Systems (JURIX 2018)*, pages 45–57. 13, 65, 149, 163

Navas-Loro, M., Satoh, K., and Rodríguez-Doncel, V. (2019b). ContractFrames: Bridging the Gap Between Natural Language and Logics in Contract Law. In *New Frontiers in Artificial Intelligence*, pages 101–114, Cham. Springer International Publishing. 12, 13, 68, 157, 158, 159, 160

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*. 97

Northwood, C. (2010). TERNIP: temporal expression recognition and normalisation in Python (MSc Thesis). Technical report. 42, 93

O'Gorman, T., Wright-Bettner, K., and Palmer, M. (2016). Richer Event Description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56. 22

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational linguistics*, 31(1):71–106. 32

Pellow, D. and Eskenazi, M. (2014). An open corpus of everyday documents for simplification tasks. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 84–93. 97

Poveda-Villalón, M., Gómez-Pérez, A., and Suárez-Figueroa, M. C. (2014). OOPS! (OntOlogy Pitfall Scanner!): An On-line Tool for Ontology Evaluation. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(2):7–34. 181, 187

Poveda-Villalón, M., Fernández-Izquierdo, A., and García-Castro, R. (2019). Linked Open Terms (LOT) Methodology. 181, 221

Prasojo, R. E., Kacimi, M., and Nutt, W. (2018). Modeling and summarizing news events using semantic triples. In *European Semantic Web Conference*, pages 512–527. Springer. 53

Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003a). TimeML: Robust Specification of Event and Temporal Expressions in Text. In *New Directions in Question Answering*, pages 28–34. 30

Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003b). The Timebank corpus. In *Corpus linguistics*, page 40. 30

Pustejovsky, J., Lee, K., Bunt, H., and Romary, L. (2010). ISO-TimeML: An International Standard for Semantic Annotation. In *LREC*, volume 10, pages 394–397. 16

Pustejovsky, J., Verhagen, M., Nianwen, X., Gaizauskas, R., Hepple, M., Schilder, F., Katz, G., Saquete, E., Caselli, T., et al. (2009). TempEval2: Evaluating events, time expressions and temporal relations. In *SemEval Task Proposal*. 52

Quaresma, P. and Gonçalves, T. (2010). Using linguistic information and machine learning techniques to identify entities from juridical documents. In *Semantic Processing of Legal Texts*, pages 44–59. Springer. 88

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann. 40

Ramakrishna, K., Guda, V., Rani, B. P., and Chakati, V. (2011). A Novel Model for Timed Event Extraction and Temporal Reasoning In Legal Text Documents. *International Journal of Computer Science and Engineering Survey*, 2:39–48. 44, 50

Ranasinghe, T., Orasan, C., and Mitkov, R. (2020). Intelligent translation memory matching and retrieval with sentence encoders. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 175–184. 124

Registry of the Court (2020). `https://www.echr.coe.int/documents/rules_court_eng.pdf`. accessed 2020-09-14. 149

Reichenbach, H. (1947). *Elements of symbolic logic*. New York:Free Press. 40

Rodríguez-Doncel, V. and Montiel-Ponsoda, E. (2020). LYNX: Towards a Legal Knowledge Graph for Multilingual Europe. *Law in Context. A Socio-legal Journal*, 37(1):1–4. 194, 196

Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T., and Bogaard, T. (2016). Building event-centric knowledge graphs from news. *Journal of Web Semantics*, 37-38:132–151. 194

Roth, M. and Lapata, M. (2015). Context-aware frame-semantic role labeling. *Transactions of the Association for Computational Linguistics*, 3:449–460. 47

Ruppenhofer, J., Ellsworth, M., Schwarzer-Petruck, M., Johnson, C. R., and Scheffczyk, J. (2006). FrameNet II: Extended theory and practice. Technical report. 32

Satoh, K., Asai, K., Kogawa, T., Kubota, M., Nakamura, M., Nishigai, Y., Shirakawa, K., and Takano, C. (2011). PROLEG: An Implementation of the Presupposed Ultimate Fact Theory of Japanese Civil Code by PROLOG Technology. In *New Frontiers in Artificial Intelligence*, pages 153–164. Springer Berlin Heidelberg. 158

Saurí, R., Batiukova, O., and Pustejovsky, J. (2009). Annotating events in Spanish. TimeML annotation guidelines (version TempEval-2010). Technical report, Barcelona Media. Technical Report BM 2009-01. 17

Saurí, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006a). TimeML annotation guidelines. *Version*, 1(1):31. 17, 18

Saurí, R., Saquete, E., and Pustejovsky, J. (2010). Annotating time expressions in Spanish. TimeML annotation guidelines (version tempeval-2010). Technical report, Barcelona Media Technical Report 2010-02. 4, 16, 105

Saurí, R., Verhagen, M., and Pustejovsky, J. (2006b). SlinkET: A partial modal parser for events. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA). 40

Scherp, A., Franz, T., Saathoff, C., and Staab, S. (2012). A core ontology on events for representing occurrences in the real world. *Multimedia Tools and Applications*, 58(2):293–331. 24

Schilder, F. (2005). Event Extraction and Temporal Reasoning in Legal Documents. In *Annotating, Extracting and Reasoning about Time and Events, International Seminar, Dagstuhl Castle. Revised Papers*, pages 59–71. 43, 85

Schmid, H. (1995). TreeTagger-a language independent part-of-speech tagger. *Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart*, 43:28. 38

Schrodt, P. A. (2012). Cameo: Conflict and mediation event observations event and actor codebook. *Pennsylvania State University*. 23

Scott, W. A. (1955). Reliability of content analysis: The case of nominal scale coding. *Public opinion quarterly*, pages 321–325. 92

Segers, R., Laparra, E., Rospocher, M., Vossen, P., Rigau, G., and Ilievski, F. (2016). The Predicate Matrix and the Event and Implied Situation Ontology: Making More of Events. In *Proceedings of the 8th Global WordNet Conference*, pages 364–372. 26

Setzer, A. (2002). *Temporal information in newswire articles: an annotation scheme and corpus study.* PhD thesis, University of Sheffield. 16

Shekarpour, S., Alshargi, F., Thirunaravan, K., Shalin, V. L., and Sheth, A. (2019). CEVO: comprehensive event ontology enhancing cognitive annotation on relations. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, pages 385–391. IEEE. 32

Sierra, G., Bel-Enguix, G., López-Velarde, G., Saucedo, R., and Rivera, L. (2018). Event Extraction from Legal Documents in Spanish. In *Workshop on Language Resources and Technologies for the Legal Knowledge Graph, LREC 2018. Miyazaki, Japan.* 49

Song, Z., Bies, A., Strassel, S., Ellis, J., Mitamura, T., Dang, H. T., Yamakawa, Y., and Holm, S. (2016). Event nugget and event coreference annotation. In *Proceedings of the Fourth Workshop on Events*, pages 37–45. 22

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In *Proceedings of the the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98. 21

Sprugnoli, R. and Tonelli, S. (2017). One, no one and one hundred thousand events: Defining and processing events in an inter-disciplinary perspective. *Natural Language Engineering*, 23(4):485–506. 31

Sprugnoli, R. and Tonelli, S. (2019). Novel event detection and classification for historical texts. *Computational Linguistics*, 45(2):229–265. 23

Steedman, M. and Baldridge, J. (2011). *Combinatory Categorial Grammar*, chapter 5, pages 181–224. Wiley-Blackwell. 41

Strötgen, J. and Gertz, M. (2012). Temporal tagging on different domains: Challenges, strategies, and gold standards. In *LREC*, volume 12, pages 3746–3753. 30, 31, 36, 80, 93, 98

Strötgen, J. and Gertz, M. (2013). Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation (LREv)*, 47(2):269–298. 133, 138

Strötgen, J. and Gertz, M. (2016). Domain-sensitive temporal tagging. *Synthesis Lectures on Human Language Technologies*, 9(3):1–151. 97, 98

Styler, W. F., Bethard, S., Finan, S., Palmer, M., Pradhan, S., De Groen, P. C., Erickson, B., Miller, T., Lin, C., Savova, G., et al. (2014). Temporal Annotation in the Clinical Domain. *Transactions of the ACL*, 2:143–154. 19, 30, 46

Swayamdipta, S., Thomson, S., Dyer, C., and Smith, N. A. (2017). Frame-Semantic Parsing with Softmax-Margin Segmental RNNs and a Syntactic Scaffold. *arXiv preprint arXiv:1706.09528*. 47

Tabassum, J., Ritter, A., and Xu, W. (2016). Tweetime: A minimally supervised method for recognizing and normalizing time expressions in twitter. *arXiv preprint arXiv:1608.02904*. 30

Tannier, X. and Vernier, F. (2016). Creation, Visualization and Edition of Timelines for Journalistic Use. In *Proceedings of Natural Language meets Journalism Workshop at IJCAI*. 46

Torraco, R. J. (2005). Writing integrative literature reviews: Guidelines and examples. *Human resource development review*, 4(3):356–367. 13

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics, Volume 2: Proceedings of SemEval 2013*, volume 2, pages 1–9. 52, 97, 98

van Opijnen, M. (2016). Towards a global importance indicator for court decisions. In *Legal Knowledge and Information Systems - JURIX 2016: The Twenty-Ninth Annual Conference*, pages 155–160. 66

Verhagen, M., Mani, I., Sauri, R., Littman, J., Knippen, R., Jang, S. B., Rumshisky, A., Phillips, J., and Pustejovsky, J. (2005). Automating Temporal Annotation with TARSQI. In *Proceedings of the ACL 2005 on Interactive Poster and Demonstration Sessions*, pages 81–84. ACL. 40, 42, 93

Verhagen, M. and Pustejovsky, J. (2008). Temporal processing with the TARSQI toolkit. In *COLING 2008: Companion volume: Demonstrations*, pages 189–192. 40

Vicente-Díez, M. T., Schneider, J. M., and Martínez, P. (2010). UC3M System: Determining the Extent, Type and Value of Time Expressions in TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, page 329–332, USA. Association for Computational Linguistics. 43

Vlek, C. S., Prakken, H., Renooij, S., and Verheij, B. (2013). Representing and Evaluating Legal Narratives with Subscenarios in a Bayesian Network. In *2013 Workshop on Computational Models of Narrative*, volume 32, pages 315–332. Schloss Dagstuhl-Leibniz-Zentrum für Informatik. 44

Vossen, P. (1998). *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Publishers, MA, USA. 38

Whittemore, R. and Knafl, K. (2005). The integrative review: updated methodology. *Journal of advanced nursing*, 52(5):546–553. 13, 14

Wu, J., Zhu, X., Zhang, C., and Hu, Z. (2020). Event-centric Tourism Knowledge Graph—A Case Study of Hainan. In *Knowledge Science, Engineering and Management*, pages 3–15, Cham. Springer International Publishing. 194

Wyner, A. Z. and Peters, W. (2010). Lexical semantics and expert legal knowledge towards the identification of legal case factors. In *JURIX*, volume 10, pages 127–136. 50

Xiang, W. and Wang, B. (2019). A survey of event extraction from text. *IEEE Access*, 7:173111–173137. 46

Zhong, X., Sun, A., and Cambria, E. (2017). Time expression analysis and recognition using syntactic token types and general heuristic rules. In *Proceedings of the 55th Annual Meeting of the ACL*, volume 1, pages 420–429. 41, 93

Abstract

Legal documents can be long, complex and difficult to understand. However, there is a strong demand for access to legal information, and thousands of documents are published every day. Although there is a multitude of institutional portals available to citizens and legal practitioners, the documents themselves are often plain texts from whom it is difficult to extract information. The retrieval of temporal information in judgments is particularly important, and the analysis of these texts often requires identifying dates and events. In fact, being able to represent a sentence as a set of relevant events would be extremely useful, as it would improve searches and facilitate the visualization and understanding of texts through summaries and timelines, among others. However, there is currently no system that facilitates the processing of temporal information in legal documents.

This doctoral thesis aims to provide a framework that addresses the problem comprehensively, proposing algorithms for the recognition of temporal expressions and events, describing a data model for their representation and demonstrating that they facilitate the retrieval of temporal information in legal texts.

The main contributions are (1) several annotated corpora in the legal domain, (2) a temporal tagger capable of processing Spanish and English texts that improves the state of the art in the legal domain, (3) an event extractor for European legal decisions that also generates a timeline, and (4) a pipeline that allows transforming European legal decisions into a set of events within a knowledge graph. For this purpose, several tools and resources have been developed, such as an ontology that allows representing a document as an aggregation of its most relevant events and its temporal annotations, or a converter between different temporal annotation formats and data conforming to this ontology. All these contributions allow to transform a legal document into an event-based representation that facilitates retrieving legal information.

Resumen

Los documentos legales pueden llegar a ser largos, complejos y difíciles de entender. No obstante, existe una fuerte demanda de acceso a información legal, y diariamente se publica una gran cantidad de documentos. Pese a que existen multitud de portales institucionales a disposición de ciudadanos y profesionales del derecho, los documentos en sí suelen ser texto plano de los cuales es difícil extraer información. La recuperación de información temporal en las sentencias judiciales es especialmente importante, y el análisis de estos textos requiere a menudo identificar fechas y eventos. De hecho, poder representar una sentencia como un conjunto de eventos relevantes sería extremadamente útil, pues permitiría mejorar las búsquedas y facilitar la visualización y comprensión de los textos mediante resúmenes y líneas temporales. Sin embargo, no existe a día de hoy un sistema que facilite el procesamiento de información temporal en documentos del ámbito legal.

Esta tesis doctoral contribuye al avance del estado del arte proporcionando un marco de trabajo que aborde la información temporal de manera integral, proponiendo algoritmos de reconocimiento de expresiones temporales y eventos, describiendo un sistema de representación de los mismos y demostrando que su uso facilita consultar información temporal en textos jurídicos.

Las principales contribuciones de esta tesis son (1) diversos corpus anotados en el dominio legal, (2) un anotador temporal capaz de procesar textos en español e inglés que mejora el estado del arte en el dominio legal (3) un extractor de eventos para sentencias europeas que genera además un timeline, y (4) un pipeline que permite transformar sentencias europeas en un conjunto de eventos dentro de un grafo del conocimiento. Para ello se han desarrollado distintos recursos, como una ontología que permite representar un documento como sus eventos más relevantes y sus anotaciones temporales, o un conversor entre distintos formatos de anotación temporal y los datos representados conforme a la ontología. Estas aportaciones permiten una representación del documento que facilita el acceso a la información.