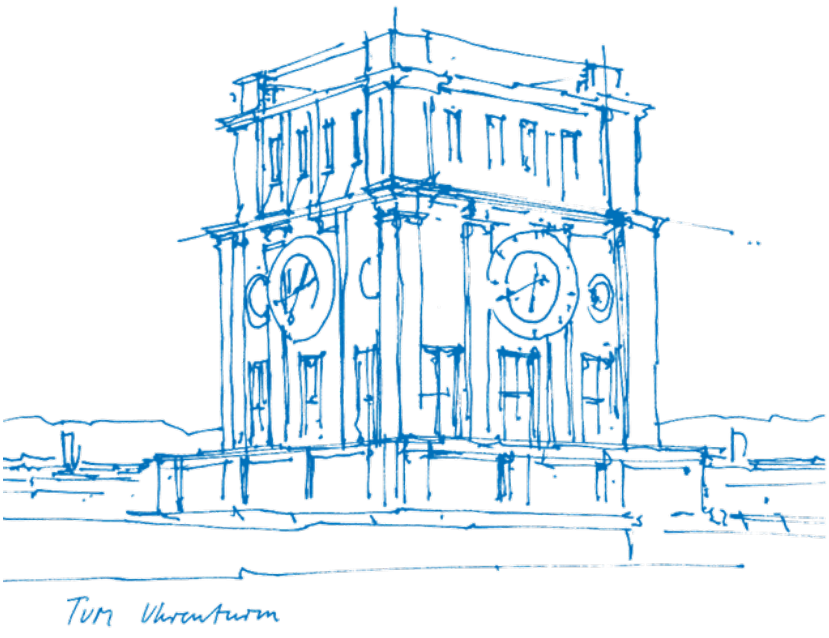
Chair of Information Systems and Business Process Management (i17)
Department of Computer Science
TUM School of Computation, Information and Technology - Informatics
Technical University of Munich

Identification and Visualization of Legal Definitions and their Relations Based on European Regulatory Documents

Anastasiya Damaratskaya

Bachelor's Thesis in Informatics

at the TUM School of Computation, Information and Technology - Informatics,
Department of Computer Science,
Chair of Information Systems and Business Process Management (i17)



Chair of Information Systems and Business Process Management (i17)
Department of Computer Science
TUM School of Computation, Information and Technology - Informatics
Technical University of Munich

Identification and Visualization of Legal Definitions and their Relations Based on European Regulatory Documents

Identifizierung und Visualisierung Legaldefinitionen und ihrer Relationen auf der Grundlage europäischer Regulierungsdokumente

Anastasiya Damaratskaya

Bachelor's Thesis in Informatics

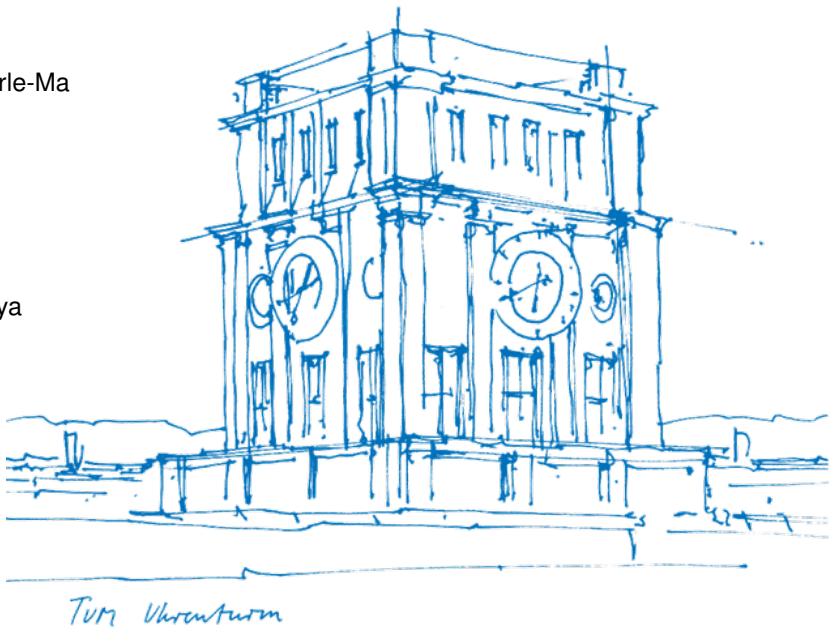
at the TUM School of Computation, Information and Technology - Informatics,
Department of Computer Science,
Chair of Information Systems and Business Process Management (i17)

Examiner
Prof. Dr. Stefanie Rinderle-Ma

Supervised by
Karolin Winter
Catherine Sai

Submitted by
Anastasiya Damaratskaya

Submitted on
15.05.2023



Declaration of Academic Integrity

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

I am aware that the thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the thesis as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future theses submitted. Further rights of reproduction and usage, however, are not granted here.

This thesis was not previously presented to another examination board and has not been published.

Garching, 15.05.2023 Anastasiya Damaratskaya

Abstract

Analyzing regulatory documents is a continuous challenge for numerous companies, especially if it is a manual process. Considering the exponential growth in legal acts, legal practitioners must invest vast amounts of time examining the legal text for relevant information. Nevertheless, the manual analysis remains susceptible to errors and misinterpretation. This thesis concentrates on semi-automating this procedure and presents an approach for extracting legal definitions and their semantic relations from European regulatory documents using natural language processing techniques. We further visualize the obtained data on the implemented web service, which serves as a practical application for the approach. Since the existing methodologies addressing legal information retrieval tasks struggle with interpreting legal text and lack semantic analysis and visualization, our method intends to cover this research gap and deepen the understanding of regulatory documents.

In order to identify legal definitions, we primarily investigated the legal acts structure that regulatory documents attempt to follow. After recognizing similar formats, we focused on a single article specifying legal terms, extracted definitions and analyzed all semantic relations occurring, such as hyponymy, meronymy, and synonymy. For this purpose, contingent upon the type of semantic relationship, we applied pattern matching and natural language processing techniques, emphasizing dependency parsing and noun phrase chunking. For visualization, the prototype collected the data into separate files and extracted sentences mentioning legal definitions for each related term. To rapidly discover these sentences in the text and obtain an overview of each term's frequency, the prototype listed the articles where the definitions occur and counted the number of retrieved sentences. Additionally, it assigned annotations to the regulatory documents, explaining the legal definitions in each paragraph to facilitate comprehension of the regulatory documents.

The evaluation outcomes demonstrated that the prototype could detect 99.9% of legal definitions and 96.7% of their semantic relations correctly, thereby delivering accurate results for the introduced approach. The study further fulfilled the established requirements intending to simplify the platform's usage. Consequently, these results demonstrate that natural language processing techniques perform well in the classification phase and are suitable for definition and relation extraction.

Keywords: *legal definitions, legal information extraction, natural language processing.*

Contents

List of Tables

List of Figures

Acronyms

FR Functional Requirement. 66

GDPR General Data Protection Regulation. 12, 13, 20, 32, 33, 38, 42, 43, 44, 51, 53, 54, 58

LIR Legal Information Retrieval. 11, 13, 16, 19, 20, 22, 23

NFR Non-functional Requirement. 65, 66, 67

NLP Natural Language Processing. 13, 14, 20, 21, 22, 23, 24, 26, 27, 28, 35, 41, 45, 47, 50, 68, 71

POS Part-Of-Speech. 21, 23, 24, 26, 32, 35, 41, 68

RQ Research Question. 14, 17, 36, 49, 57

Glossary

CELEX number is the unique reference number assigned to the EU document. 8, 17, 28, 30, 31, 50, 51, 55, 56, 58, 64, 65, 66

EUR-Lex is the official website of European Union law and other public documents of the European Union. 11, 14, 28, 30, 50, 51, 55, 57, 58

hyponymy is a semantic relationship between words in which the meaning of one word is included in the meaning of another word. 13, 14, 28, 41, 42, 43, 44, 45, 47, 48, 49, 50, 53, 54, 55, 57, 68

legal definition determines the specific lexical term used within legal texts' discourse utilizing normative rules [1]. 6, 11, 12, 13, 14, 15, 16, 17, 19, 23, 24, 25, 28, 29, 30, 31, 32, 35, 36, 40, 44, 45, 47, 48, 49

lemma is a set of lexical forms with the same stem, major part-of-speech, and word sense [2]. 21, 32, 35

meronymy is a semantic relationship between words in which the meaning of one word is a part of another word. 14, 28, 41, 43, 44, 45, 47, 48, 50, 53, 57, 68, 69

noun phrase chunk identifies a noun phrase and its associated modifiers within a sentence. 7, 21, 41, 42, 44, 45, 46, 47, 48, 53

synonymy is a semantic relationship between words in which words have a similar meaning. 14, 28, 41, 43, 45, 50, 57, 68, 69

token is a sequence of characters or a meaningful unit of text that is processed as a single entity. 20, 21, 35, 48, 61

Introduction

Regulatory documents hold considerable importance for individuals, organizations, and legal professionals by protecting and safeguarding their interests. Nonetheless, the legal acts use complex and domain-specific terminology, complicating the manual process of analyzing legal text and leading to a potential misinterpretation of the content. To overcome these challenges and accelerate the process of detecting relevant data in legal documents, researchers have been progressively addressing Legal Information Retrieval (LIR) tasks and developing systems to obtain relevant information from the legal domain. This thesis presents a prototype capable of identifying, extracting, and visualizing legal definitions and their semantic relations from European regulatory documents since previous approaches for legal definition extraction barely considered semantic relation extraction.

This chapter clarifies the motivation and problem statement while presenting a comprehensive list of the research questions addressed in the thesis. Furthermore, we explain the proposed contribution, describe the research methodology, and outline the evaluation methods. Finally, we provide a brief overview of the thesis structure.

Motivation

Nowadays, numerous companies constantly face the problem of rising compliance requirements derived from the IT environment due to digital business activities and processes [3]. The task of business process compliance is to ensure that the organization's business operations comply with the regulatory laws affecting the organization. If the organization fails to achieve compliance, this can result in financial penalties and a loss of investors. Thus, companies attach significant importance to obeying regulatory laws and reviewing the current updates of legal documents to provide transparency and more efficient operations of their businesses [4]. According to the statistics for the year 2022, around 24 622 regulatory documents were made available on EUR-Lex[1], while the number of visitors using the website lies in 46 076 161 out of 56 545 602 total visits[2]. Consequently, corporations must invest substantial amounts of time and effort searching through lengthy legal acts, primarily through manual means, to discover relevant information. The most considerable disadvantages of the manual process lie in time consumption, costs, and fallibility since the chance to overlook crucial information or misinterpret the content is pretty high. Additionally, misjudging

[1] https://eur-lex.europa.eu/statistics/2022/eu-law-statistics.html
[2] https://eur-lex.europa.eu/statistics/2022/usage.html

or slipping up by observing regulations can result in the organization's forfeiting a large amount of money. As stated by the General Data Protection Regulation (GDPR)[3], infringements of the provisions shall be subject to administrative fines up to 10 000 000 EUR-20 000 000 EUR[4], which may lead to the possible bankruptcy of a company.

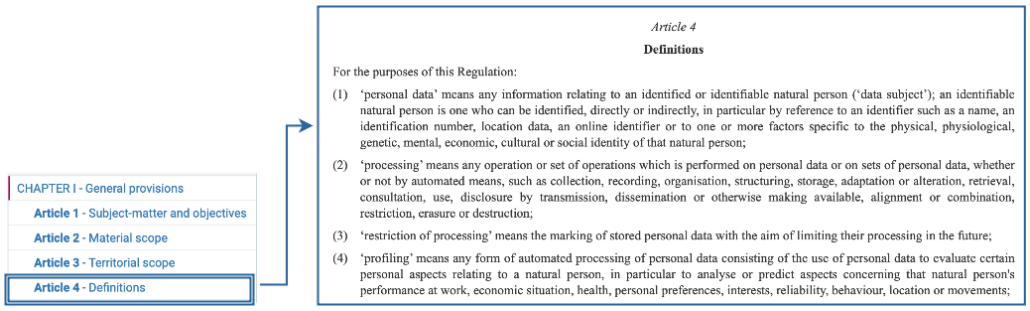


Figure 1
Illustration of the classic structure of regulatory documents. The article explaining legal definitions is commonly located at the beginning of the document, e.g. in the first chapter (example: GDPR).

In this thesis, we focus on digitizing business process compliance and facilitating the interpretation of the legal text by extracting legal definitions and their semantic relations from regulatory documents. Legal definitions determine the specific lexical terms used within legal texts' discourse utilizing normative rules [1] and are described as a rule at the beginning of each legal act (Fig. 1). These terms are not entirely used for precise and effective communication but ensure the correct interpretation of the legal text [5]. They further aid in comparing the meaning of legal definitions used in other acts, which possibly describe the same concept but with different terms. An example of the first legal definition characterized in GDPR:

> *"**'personal data'** means any **information** relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;"* (GDPR, Article 4, Point 1)

[3] `http://data.europa.eu/eli/reg/2016/679/oj`
[4] `http://data.europa.eu/eli/reg/2016/679/oj`, Chapter VIII, Article 83

Besides providing the meaning of the legal term, this definition further contains a semantic relation between the legal term and the word *"information"* called *hyponymy*, which indicates to which broader category the term refers and deepens the understanding of its meaning. Hyponymy can be beneficial for legal practitioners in diverse manners by providing more accurate distinctions between legal concepts, thereby enhancing legal reasoning and drafting. In general, a semantic relation analysis improves comprehension of the intended directive, infers implicit information, resolves ambiguities in language, and therefore requires further emphasis in addressing LIR tasks.

By semi-automating the procedure of deriving definitions and their semantic relations, both jurists and non-jurists can succeed in raising the efficiency in analyzing regulatory documents for relevant information and eliminating any ambiguity or misunderstanding, reducing the likelihood of legal disputes. In comparison to the existing approaches concentrating on the automatic processing of the legal content applying various Natural Language Processing (NLP) and deep learning techniques for LIR [6], the tool handles legal definitions specified in each regulatory document and their semantic relations in order to make a document more accessible and easier to comprehend. Furthermore, it facilitates the utilization of information through visual representation, for instance, in the form of annotations attached to the existing legal document. An example of how the prototype handles regulatory documents, in this case by processing GDPR, is presented in Figure 2.

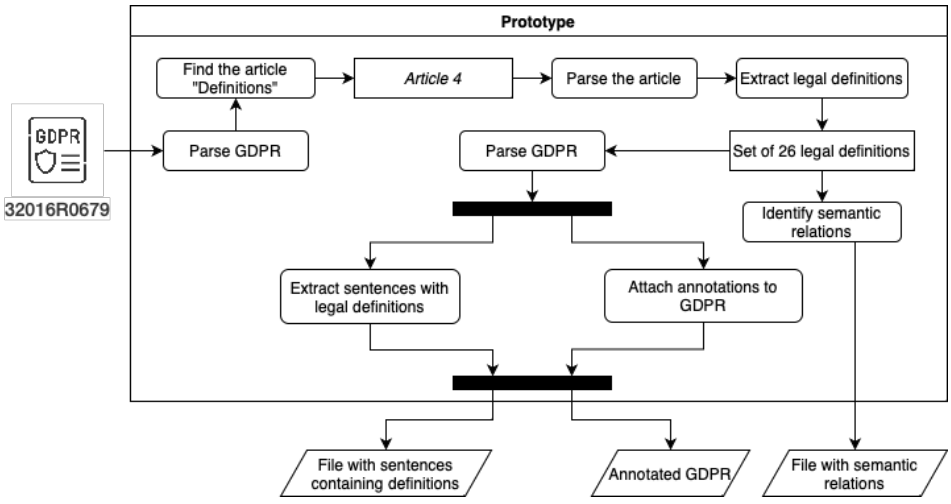


Figure 2
Using GDPR as an example input of regulatory documents, we depicted the process of analyzing the document for legal definitions and identifying semantic relations, along with providing an overview of the expected outputs.

Research Questions

Intending to extract and visualize legal definitions and their semantic relations from the processed regulatory document, we have identified four research questions that help with achieving these objectives, with particular emphasis on the first three of them:

RQ1 How to identify definitions from regulatory documents?

RQ2 How to identify semantic relations among definitions?

RQ3 How to visualize definitions and their relations?

RQ4 What additional information extracted from regulatory documents would be beneficial for the user?

To address these questions, this thesis proposes a research methodology that involves implementing and evaluating an automatic prototype capable of extracting and visualizing legal definitions and their semantic relations from regulatory documents using NLP techniques. For visualization, the prototype collects the obtained data into separate files and afterwards offers these files to users. Additionally, the implemented web service outlines the domain-specific terms most frequently used in a document, in the form of statistics, along with their exact location for the users. This information facilitates legal search and discovery of critical information in an accelerated manner.

Contribution

Earlier approaches concentrating on the definition and relation extraction required the involvement of experts in the legal field, which demands a considerable amount of human capital and time investment. This thesis introduces an automated prototype capable of extracting legal definitions and their corresponding semantic relations (i.e., hyponymy, meronymy, and synonymy) from the Regulations of the European Parliament and of the Council published on EUR-Lex[5]. We consider this tool beneficial for law experts and the general public, and the approach can be expanded or applied to a wider variety of European regulatory documents. Therefore, the contributions of this thesis are as follows:

- We present an approach for legal definition and semantic relation extraction and visualization, including obtaining text segments containing definitions and annotating the entered regulation by applying NLP techniques.

[5] `https://eur-lex.europa.eu/`

- We implement an intuitive web service that adopts the proposed approach to enable users to access the extracted information referring to legal definitions.
- We additionally discover the frequency of specific definitions and their locations to accelerate the legal search and provide insight into the regulation's objective.

Methodology

This section summarizes the design-science research guidelines proposed by Hevner et al. [7].

Design as an Artifact

A purposeful IT artefact of this thesis is an implemented prototype for analyzing regulatory documents, specifically European regulations, and extracting legal definitions to return an adjusted regulation with annotations in an HTML format. Additionally, the prototype supplies users with three text files: the first lists all extracted legal terms with their explanations, the second one contains text segments where the obtained definitions occur, and the last one collects the derived semantic relations among definitions, as demonstrated in Figure 3. Both jurists and non-jurists should be able to use the proposed prototype, which serves as a solver for the problem of processing long regulatory documents manually and misinterpretation of regulatory documents.

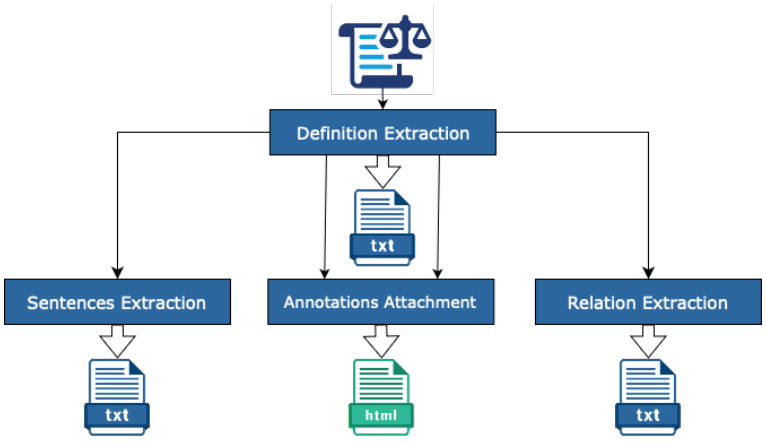


Figure 3
The suggested contribution is an approach for extracting legal definitions, text segments mentioning them, and semantic relations and returning a text file with the obtained information. Furthermore, the explored explanations of terms are attached to the initial regulatory document and returned as an HTML document.

Problem Relevance

The critical problems addressed in this thesis are the time consumption, costs, and fallibility of the manual examination of regulations for each company that needs to constantly handle regulatory documents in order to stay updated about current changes in laws and policies. The goal is to semi-automate the processing of regulatory documents, aid the manual interpretation, prevent high expenses, and save a substantial amount of time. Moreover, one of the open issues in LIR systems is the false interpretation of regulatory documents [6], which is further addressed by obtaining legal definitions and their semantic relations. This knowledge guarantees a better understanding of the regulation's scope and objective.

Design Evaluation

To demonstrate utility, quality, and efficacy, we establish functional and non-functional requirements upon which the evaluation of the prototype is based. Intending to investigate the correctness of the implemented prototype, we calculate the resulting sets' precision, recall and F-measure with legal definitions, their semantic relations, extracted text segments, and the regulation containing annotations. For the simulation, we execute the tool with 18 Regulations of the European Parliament and of the Council corresponding to various topics.

Research Contributions

The main contributions of this research are listed in section 1.3. The discussed approach enables the solution to the mentioned problems, which were not fully addressed by any other already present prototype as demonstrated in Chapter 2.

Research Rigor

We investigate the most common measures for evaluating information retrieval systems with the purpose of being mathematically rigorous in determining whether the prototype works accurately, and, as a result, calculate the precision, recall, and F-measure of the approach.

Design as a Search Process

The appropriate research means, which Hevner et al. [7] defines as a set of actions or resources for constructing a solution, ends, which are goals and constraints, and laws considered as the environment's uncontrollable forces are listed as follows:

- **Means:** text segments from the article *"Definitions"* representing legal terms and their explanations.
- **Ends:** resulting documents with retrieved data.
- **Laws:** regulation's text, where text segments can only be slightly changed, preventing any modifications of the structure and meaning of the sentence.

Communication of Research

For a technology-oriented audience, all details of the implementation and evaluation of the described prototype are documented for possible reproduction or extension. A management-oriented audience can find critical information about the prototype and how to apply it in the README file. Moreover, the tool provides a list of the advantages of using the prototype instead of analyzing regulatory documents manually.

Evaluation

The evaluation concentrates on estimating the approach's correctness in addressing RQ1-RQ3 and fulfilling the functional and non-functional requirements formulated in accordance with the creation of a user-friendly web service and a qualitative visualization of the detected information referring to RQ3 and RQ4.

To evaluate the precision of the prototype, we calculate the characteristic measures for information retrieval tasks such as precision, recall, and F_1. This assessment demonstrates the improvement over the previous approaches focusing on extracting legal definitions and their semantic relations, as well as estimates the performance of the introduced tool for achieving a satisfactory level. We evaluate definition extraction and relation extraction tasks together since the correctness of the identified semantic relations directly dependents on the results of definition extraction. Similarly, we evaluate sentence extraction and assignment of annotations collectively as both are implemented in one algorithm. We further investigate the web service's response to the invalid input by entering nonexistent CELEX numbers and numbers of unsupported regulatory documents.

The evaluation of the implemented prototype includes an assessment of both its functional and non-functional requirements in order to determine the level of service quality provided. This analysis involves a comprehensive examination to ensure all requirements for the main functionality are fulfilled accordingly, such as handling the incorrect input, identifying and processing a regulation

if present, and outputting the results to visualize the relevant information. Furthermore, assessing non-functional requirements offers valuable insights regarding the effectiveness of the visualization for end-users and its intuitive usability in the practical application.

Structure

This bachelor thesis is structured as follows: after Chapter *Introduction*, Chapter 2 establishes the existing methodologies for legal definition extraction along with semantic relation extraction and points to the current research gaps in legal information retrieval. Chapter 3 explains the approach we developed, providing a pseudo-code for each addressed task, followed by implementation details of the prototype described in Chapter 4. In Chapter 5, we demonstrate the clarifying statistics and the statistical analysis results of information extraction evaluation, as well as itemize and analyze the requirements for the introduced prototype. In Chapter 6, we summarize and explain our research findings, including listing the perceived limitations and possible future improvements. Finally, the established objectives are reflected upon in Chapter 7, which serves as the conclusion of this thesis research.

Related Work

Legal Information Retrieval (LIR) is a discipline that intends to extract relevant information from legal text. Since the quantity of regulatory documents nowadays increases exponentially, various developed systems aim to automate retrieval. In consideration of this evidence, we performed a systematic search of the literature (Fig. 1), conducted in DBLP[6] and Google Scholar[7] databases, where after reviewing the articles, we selected the papers containing search string in the title, abstract, or keywords for the further examination. We started with analyzing the studies for state-of-the-art of the LIR systems and investigated the current research gaps in the field. Afterwards, we narrowed the search focusing primarily on the current approaches in extracting legal definitions. Since only a few of them briefly considered semantic relations, we further examined methodologies addressing the relation extraction task.

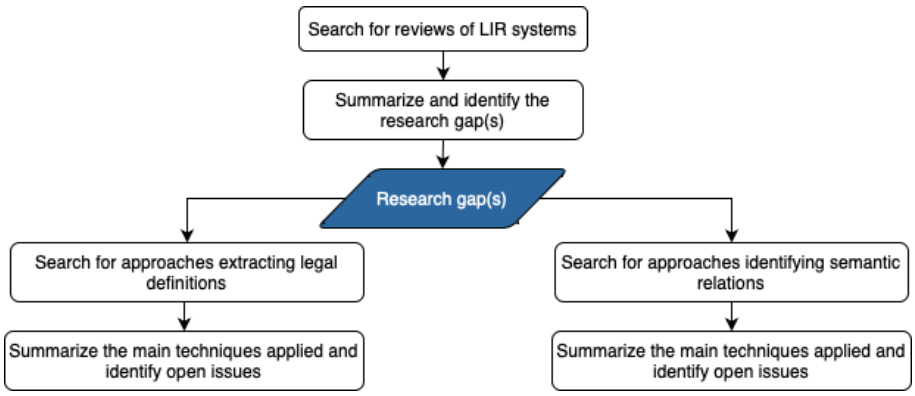


Figure 1
Flowchart depicting systematic research, beginning with a big picture and analyzing the issues of existing approaches focusing on addressing LIR tasks and proceeding with a comprehensive investigation on present solutions of definition and relation extraction.

This chapter provides an overview and explanation of the main terms concerning the introduced approach and techniques applied in LIR systems for processing legal text, along with describing the results of the systematic research.

Definition of Terms

This section explains all terms important for understanding the presented approach and provides examples of the use for a better comprehension of the terms' meaning.

[6] https://dblp.org
[7] https://scholar.google.de

Text Mining vs. Natural Language Processing

Due to the fact that we are dealing with natural language and aim to extract relevant information from the text, it is necessary to distinguish between text mining and NLP.

Definition 2.1.1 Text mining is the discovery and extraction of interesting, non-trivial knowledge from free or unstructured text. [8]

Definition 2.1.2 Natural Language Processing (NLP) is the attempt to extract a fuller meaning representation from free text. It typically makes use of linguistic concepts such as part-of-speech and grammatical structure and has to deal with anaphora and ambiguities. [8]

Since the focus of this thesis lies in improving the interpretation of regulatory documents, we adhere to NLP and apply its techniques in the demonstrated approach.

Legal Information Retrieval

Both the results of the systematic search and the presented approach concentrate on the legal field, addressing the LIR task.

Definition 2.1.3 Legal Information Retrieval (LIR) is a discipline of information retrieval applied specifically for the legal domain that intends to model information search and to identify relevant information for legal practitioners' jobs.

Text Processing Techniques

Various techniques are applied in order to retrieve the information from the text. To clarify the meaning of the terms, we illustrate the usage of them with the examples based on the text:

> *"This Regulation lays down rules relating to the protection of natural persons [...]."*
> (GDPR, Article 1, Point 1)

Definition 2.1.4 Tokenization refers to a process of replacing sensitive data with one-of-a-kind identification symbols (tokens) that maintain all of the material's critical information while providing its security [9]. Usually, the tokens represent words, numbers, or punctuation marks extracted from the text.

After tokenization: 'This', 'Regulation', 'lays', 'down', 'rules', 'relating', 'to', 'the', 'protection', 'of', 'natural', 'persons', '.'.

Definition 2.1.5 Part-Of-Speech (POS) tagging is a NLP method responsible for positioning words in a text into different parts of speech based on the context and definition of the words. [9]

After tokenization and POS tagging: 'This' →DET, 'Regulation' →PROPN, 'lays' →VERB, 'down' →ADP, 'rules' →NOUN, 'relating' →VERB, 'to' →ADP, 'the' →DET, 'protection' →NOUN, 'of' →ADP, 'natural' →ADJ, 'persons' →NOUN, '.' →PUNCT.

Definition 2.1.6 Dependency parsing is a form of syntactic parsing of natural language, which examines sentences by automatically constructing representations of their semantic structure. [10]

After tokenization and dependency parsing: 'This' →det, 'Regulation' →nsubj, 'lays' →ROOT, 'down' →prt, 'rules' →dobj, 'relating' →acl, 'to' →prep, 'the' →det, 'protection' →pobj, 'of' →prep, 'natural' →amod, 'persons' →pobj, '.' →punct.

Definition 2.1.7 Lemmatization refers to a process of determining whether two words have the same root, despite their surface differences, by reducing them to their base, known as a "lemma". [2]

After tokenization and lemmatization: 'This' →this, 'Regulation' →Regulation, 'lays' →lay, 'down' →down, 'rules' →rule, 'relating' →relate, 'to' →to, 'the' →the, 'protection' →protection, 'of' →of, 'natural' →natural, 'persons' →person, '.' →..

Definition 2.1.8 Noun phrase chunking is a NLP technique that involves grouping a consecutive sequence of words into a noun phrase chunk label. [11]

After noun phrase chunking: 'This Regulation', 'rules', 'the protection', 'natural persons'.

Definition 2.1.9 Stop words are worthless and insignificant standard terms from the tokens streams, such as articles, prepositions etc., which should be removed in order to normalize the text. [12]

After noun phrase chunking and removing the stop words: 'Regulation', 'rules', 'protection', 'natural persons'.

Legal Information Retrieval and Open Issues

The main challenge of LIR lies in a mismatch between the user's vocabulary and the legal domain language. Moreover, identifying which information is required for the user is likewise complex. Ibrihich et al. [12] demonstrated modeling and simulation approaches to explain information retrieval basics and compared the existing techniques used in information retrieval. The authors listed various stages of text processing which are commonly fulfilled by an information retrieval engine and can be considered an excellent foundation for any approach addressing LIR tasks.

Investigating existing methods for LIR systems and possible research gaps, Sansone et al. [6] provided a comprehensive overview of artificial intelligence approaches for the legal domain, concentrating on LIR systems applying natural language processing, machine learning, and knowledge extraction techniques. They further explained in their studies the tasks referring to LIR and current issues of the existing approaches, such as the relation of the relevance concept to legal interpretation and the problem of identifying frequently modified regulatory documents. Moreover, the authors highlighted the need to manage the regulatory inter-relationships and provide semantic analysis instead of only syntactic-grammatical. Further challenges in the legal domain are open to developing a LIR system capable of improving the understanding of legal documents and their structure, as well as supplying a correct interpretation and summarization of the legal text.

Another review encapsulating the knowledge on emerging Intelligent Information Retrieval practices in the legal domain is presented in [13], further offering background and research considering techniques for information retrieval in regulatory documents, documenting the current state-of-the-art, and identifying its main weaknesses. Gomes et al. mentioned techniques for revealing understandings and patterns that can be used to support decision-making, which are semantic web, text mining, and NLP.

In [14] Locke et al. examined methods for case law retrieval from the past 20 years and discussed the problems and challenges facing evaluating these systems. Most of the current approaches turned out to be theoretical, and the evaluation of the algorithms is more frequently empirical rather than a more grounded analysis applied, such as comparisons with baselines, human evaluation and

statistical significance testing. A different area that needs to be investigated is semantic search since inspecting a collection solely for similar factual situations can be beneficial for legal practitioners, as well as a search for regulatory documents that are logically similar but not factually relevant. The significance of the meaning of words must be taken into account since it offers an accurate interpretation of the legal document. Therefore, the ability to search for legal information that explains the word's meaning is further useful for improving the semantic understanding of the legal text.

Legal Definition Extraction

Summarizing the state-of-the-art approaches for LIR systems, they intended to solve the task of identifying the most relevant information in the legal document, which impacts the correctness of the legal interpretation. Some of these approaches further address the problem of an accurate interpretation, which is strongly related to the relevance concept. Nevertheless, the desired goal can be achieved by extracting legal definitions and terms from the legal document, as European regulatory documents often describe legal definitions in order to guarantee their common understanding by all parties [15]. The definitions serve the correct interpretation of the legal text and are not entirely for precise and effective communication [5]. Throughout semi-automating the procedure of deriving definitions and their relations, both jurists and non-jurists can succeed in raising the efficiency in analyzing regulatory documents for relevant information and interpreting it accurately.

Ferneda et al. [16] presented an approach for automating the process of extracting legal terms based on a variation of an automated technique of NLP of Brazilian Portuguese texts, focusing on the more general issue of establishing a glossary from domain-specific texts that embrace definitions amongst their content. The authors explored in depth the meaning of the term *definition* and settled on Aristotle's explanation of the *genus et differentia* type of definition, which interprets a term *(definiendum)* by its kind *(genus)* and distinguishing characteristics *(differentia)*. This definition variant supports hyponym-hyperonym relations as *genus* refers to the more general term (hyperonym) and *definiendum* to the specific one (hyponym), although the approach was not suitable for detecting semantic relations among definitions. Considering applied technologies, they used NLTK techniques for Part-Of-Speech (POS) tagging and tokenization, as well as regular expressions for text segmentation in paragraphs. After segmenting, a feature extraction function, which consists of manually defined rules established on observing some samples present in the training corpus,

returned a numerical value for each text element indicating whether a processed text segment is a definition. The obtained results of definition extraction are 75.6% precision, 69.6% recall and 72% F-measure, and thus, a manual review process is still necessary afterwards.

Hwang et al. [17] investigated legal and domain-relative term extraction on Chinese law text, building afterwards a law ontology. They further applied NLP techniques with the support of data mining to extract legal keywords and their definitions automatically. Similar to the previous approach, the system first divided sentences into segments and then attached the POS tag to each of them. To facilitate the finding of legal terms, the definition patterns were examined by law experts in the text and transformed into regular expressions. The approach extracted all pairs (keyword, definition), associating them with one or more originating statutes. Unfortunately, some of the pairs were eventually not referring to legal definitions, but were still extracted and handled as so, and therefore the results still need manual verification by experts.

Another automatic method for extracting legal terms and their explanations from a Japanese statutory corpus is described in [18]. Nakamura et al. addressed the disunity for word selection in the translation process. They developed an approach capable of constructing a legal ontology consisting of sufficient entries and semantic relations between them for translation. After completing a linguistic analysis, the authors concluded that legal documents are likely to use equivalent expressions and arranged that surface pattern rules are acceptable for term extraction implemented in the approach. However, the analysis of the results revealed that some of the rules were still error-prone.

To explore the use of legal definitions in Indonesian regulations, Amaludin et al. [5] proposed a text mining solution, where the appearance of terms is first analyzed and then summarized into regular expressions for a faster search. Due to the fact that regulations have a standardized structure and legal definitions are consistently defined in the general provision of the regulations, the approach detects the general provision part of each regulation and splits it with the sentence tokenizer into the segmentation of sentences. Afterwards, the tool analyzed each sentence for the legal definition pattern and extracted terms, alias, and definitions. However, using regex patterns was not fully successful, and the tool could not recognize some terms as definitions. Additionally, the correctly identified components had often misspelled words, missing letters, incorrect word order, and mixed words, which is why further data cleaning was required. Afterwards, the authors used a density-based clustering algorithm to group similar words into groups, which then labelled each legal definition.

In [19] the authors used the equivalent explanation of the term *definition* as in [16] and developed an approach for automatic extraction of legal definitions similar to [20], [21], where they formulated a set of typical patterns of sentences containing definitions and searched the texts for occurrences of these patterns. Depending on the pattern, the definitions extraction was evaluated individually with a mostly high recall value (94% and above). These results demonstrated that using surface pattern-matching methods can help automate the process of legal definition extraction.

Focusing on German laws and cases, Waltl et al. [22] demonstrated the rule-based approach for extracting legal definitions and other relevant semantic information, such as the year of dispute. To determine definitions and define contexts, the authors created a taxonomy differentiating between legal definitions in a narrow sense, contexts that extend definitions, and interpretation of legal terms.

Maat et al. [23] investigated the meaning of definitions in regulatory documents deeper, further pointing out the similarity of text patterns by describing the legal term. The implemented classifier uses pattern matching with the support of regular expressions and tries to match each sentence to the corresponding pattern. Evaluation results proved that the approach performed well by correctly classifying 91% of all sentences and 81% of all lists. Moreover, it shows that most laws apply only limited patterns. The accuracy of the classifier is though directly dependent on the patterns set; even supposing that the amount of variation in legal texts is restricted, it can drop if the set misses the values. Additionally, the performance of the approach decreased by processing lists in comparison to sentences since the classifier relied on the first sentence of the list and classified it, although list items belonged to another pattern.

Semantic Relation Extraction

After analyzing the approaches capable of extracting legal definitions, we further investigate classifiers addressing the relation extraction task, which is responsible for finding relations between semantic concepts in the text [2]. Korger et al. [24] presented a rule-based approach for relation extraction, named entity recognition and semantic concept extraction, using spaCy for natural language operations and Apache UIMA for visualization and annotation standardization. Their case study observed overlapping classes, incidents and measures, which complicated the separation of the received information. The classifier also combined different methods like the co-occurrence of entities, textual patterns, and syntactic patterns to detect semantic relations. As a result, the

authors assumed that lexical rules for identifying sentences containing relevant semantic concepts and relational information could improve the performance.

Another machine learning-based technique to automatically extract semantic knowledge from the legal text is described in [25], which uses syntactic dependencies between extracted terms together with a syntactic parser. Boella et al. assumed that a semantic tag can be distinguished by limited sets of syntactic contexts so the relation extraction task is identified as a classification problem where each term occurring in the sentence has to be associated with a specific semantic label given its syntactic dependencies.

Fundel et al. [26] criticized pattern-based extraction approaches since they achieve significantly lower recall and developed the rule-based classifier called RelEx that extracts biomedical semantic relations starting from long and complicated free-text sentences. The prototype applied POS-tagging and noun chunking using already existing tools for the sentences, which then were submitted to the dependency parse tree for word positions assignment. The candidate relations are created by extracting the paths connecting pairs from the dependency tree and contain only relevant terms referring to the possible relation. Concluding, the tool achieved higher performance results than existing approaches addressing relation extraction task but is fully dependent on the applied publicly available preprocessing tools.

A comparable approach is described in [9], where semantic relations are obtained from the corpus of news recommendations. Starting with extracting a sentences set depending on delimiters like white space, commas, and semicolons, the approach tokenizes the text, applying POS tagging and stop words on it, thereby identifying the occurring semantic relations in the sentence. The studies demonstrated that NLP techniques are suitable for relation extraction and provide good performance in the classification phase.

A proper summary of resources and tools for relation extraction is presented in [27], which investigates both machine learning and rule-based techniques, as well as analyzes the impact of the different levels of linguistic knowledge for the various approaches. Analogous to previous approaches, the classifier begins with the preprocessing phase, where the sentences are split, tokenized, lemmatized, and POS tagged. After that, the named entity recognition and dependency parsing are applied to the text segments, and depending on the strategy (distant supervision, supervised classifiers, novel

rule-based), the relevant information is extracted. Similarly, the authors in [28] developed a system called RelExt that automatically recognizes highly relevant pairs of concepts connected by a relation over concepts from an existing ontology. The approach extracts relevant verbs, which usually express the relation and interaction between terms, and their grammatical arguments from a domain-specific text collection and computes semantic relations via merging linguistic and statistical processing.

After analyzing the existing systems, we conclude that most of them applied similar NLP and pattern matching techniques for the addressed tasks, which are considered to perform well according to the results. Building upon this foundation, we further used the technologies in our proposed approach, combining definition and relation extraction in one method. Moreover, the limited amount of the current approaches visualized the obtained data, although the visualization supports decision-making and simplifies the identification of key findings. Accordingly, we proposed a prototype capable of visualizing the extracted information and providing the files with data for users to download.

Solution Design

In this chapter, we introduce a prototype capable of extracting legal definitions and their semantic relations from European regulatory documents, specifically from Regulations of the European Parliament and of the Council published on EUR-Lex, based on NLP techniques. Figure 1 illustrates step-by-step the functioning methodology of the approach, beginning with entering the CELEX number and finishing with the visualization. It further collects the extracted information in files containing legal terms and text segments where they occur, as well as a list of all identified relations such as hyponymy, meronymy, and synonymy. Moreover, the tool provides an annotated regulation, highlighting the first occurring legal definitions in each text segment and displaying a corresponding explanation for better usability.

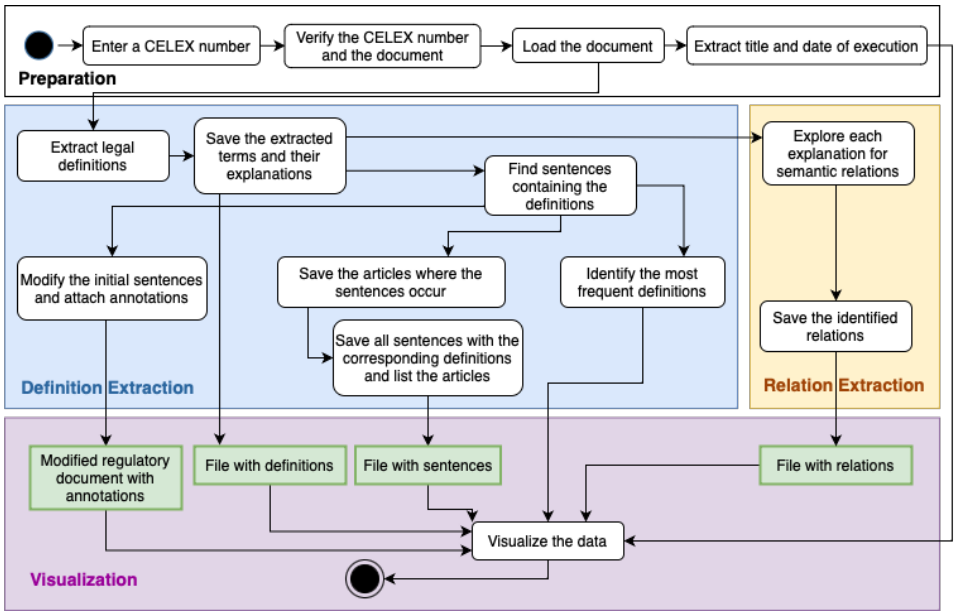


Figure 1
Flowchart representing how the introduced approach processes an entered regulatory document in each stage.

The research methodology of this thesis consists of three phases, namely **definition extraction**, **relation extraction** and **visualization of extracted information**, where each phase addresses at least one of the research questions. However, before starting with the extraction, it is essential to investigate the structure of regulatory documents to identify the most relevant parts of the document and improve the efficiency of the introduced approach.

Regulatory Document Structure

Aiming to simplify the retrieval process and comprehend the content and meaning of the regulatory documents, we analyze the components and rules concerning legal acts. Typically, these documents have a defined structure, including a title, preamble, enacting terms with articles, and concluding annexes, as demonstrated in Table 1.

The joint practical guide [29] comprehensively describes the structure of the legal acts and mentions that each legal document type has its standard presentation and standard formulations (point 2.1). According to this guideline, regulatory documents must be drawn up in a clear, simple, and precise manner (point 1.1), in conformity with the uniform principles for the presentation and development of legislative drafting for citizens and economic operators to be able to establish their rights and obligations and for the courts to enforce them (point 2.2.1). Binding acts should prescribe rules and contain provisions with information, such as the scope and the legal definitions, necessary for correct understanding and application of the rules (point 12.1).

	Description
Title	Collection of all the information necessary to identify the act
Preamble	Citations, recitals and solemn forms
Enacting terms	Legislative part of the act including articles
Annexes	Integral part of the act containing technical material

Table 1
Essential elements of the structure of legal acts according to point 7.

Whereas the consistency of terminology must be maintained so that equal terms constantly express the same concepts (point 6.2), sentences in legal acts should declare only one main idea. Since they comprise more than one sentence, articles must group various ideas with a logical link between them (point 4.4). The grammatical relationship (e.g., a relation of adjective to one or several nouns) in each sentence between parts of speech must also be clear (point 5.2.3). The use of abbreviations should be limited and further depend on the potential addressees who are familiar with them, or the meaning of abbreviations must be clearly explained the first time they are mentioned (point 4.7).

To avoid faulty interpretation of legal acts and improve general perception, the terms should be defined (point 6.2.3) and their definitions respected throughout the act (point 6.4). If the terms are ambiguous, they should be described in a single article called *"Definitions"* right at the beginning

(point 14), following the first article, establishing subject matter and scope (point 13.2), as illustrated in Table 2. If the separate article containing defined terms does not exist, then regulatory documents include them in the first article together with the scope (point 13.4). Other than that, legal definitions are required to not contain autonomous normative provisions since it can lead to misinterpretation (point 14.4).

Enacting terms
Subject matter and scope
Definitions
Rights and obligations
Provisions delegating powers and conferring implementing powers
Procedural provisions
Measures relating to implementation
Transitional and final provisions

Table 2
Point 15 mandates the inclusion of, at the minimum, listed articles in enacting terms to guarantee the fullness of the legal act.

Based on these findings, we assume that most regulatory documents keep the structure of enacting terms and include a separate article with legal definitions that the approach aims to extract.

CELEX Number

Since the presented approach takes as an argument a CELEX number of legal documents, we further analyze the compound of this number. For this purpose, we examine another guideline[8] provided by EUR-Lex, explaining constituent parts of a unique CELEX number. The CELEX number consists of four units:

- First unit (1 digit or 1 letter): Sector
  - In general, EUR-Lex categorizes 12 sectors (treaties, case-law, etc.); however, in this work, we concentrate on regulations that refer to the legislation sector **3**.
- Second unit (4 digits): Year
- Third unit (1 or 2 letters): Document type
  - L = Directives
  - R = Regulations
  - D = Decisions

[8] https://eur-lex.europa.eu/content/tools/eur-lex-celex-infographic-A3.pdf

- Forth unit (4 digits): Document number

These findings are the foundation for validating the CELEX number and the referring regulatory document for the article listing legal definitions before starting the definition extraction phase.

Definition Extraction

After the regulatory document is tested for containing an article with legal definitions, usually called *"Definitions"*, the tool begins with the definition extraction phase. With the purpose of improving efficiency, the prototype does not analyze the whole document and legal terms found in the whole text. Instead, it focuses on obtaining the definitions from this specific article, as demonstrated in [5]. Afterwards, each text segment extracted from the article is examined for legal definition patterns, such as including apostrophes and a specific verb indicating the explanation of the term. If the text segment matches the pattern, the approach separates the definition and explanation parts from each other in order to detect possible multiplicity in both fragments. Figure 2 illustrates all these stages in a precise way. Subsequently, the results are stored as sets and combined as a list for further phases.

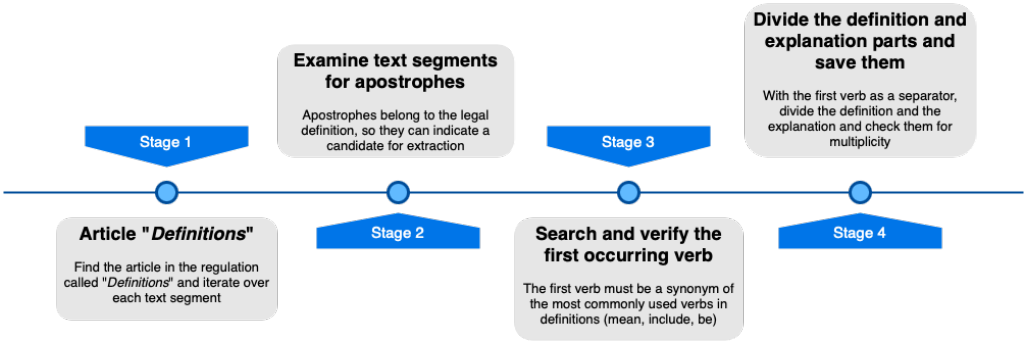


Figure 2
Four stages represent the algorithm of definition extraction from European regulations, starting from detecting a particular article with definitions and extracting each sentence to processing the data and dividing the obtained definitions into terms and explanations.

In this section, we first precisely study how legal definitions are formulated in the regulatory documents and, based on the outcome, refer to a text segment as a legal definition. Afterwards, we describe how the annotations are attached to the legal document and how the prototype determines text segments containing legal terms and extracts them.

Punctuation Marks in Legal Definitions

Punctuation plays an essential role in understanding the syntax of legal definitions and identifying semantic relations among them. The most common punctuation marks influencing the presented approach are:

- **Apostrophes:** Identify the start and the end of each legal term, except a definition itself contains an apostrophe.
  - Example: *"'the Union's 2030 targets for energy and climate' means the Union-wide binding target [...]"*[9]
- **Semicolon:** Shows the end of the independent clauses. In the approach, we examine only the first clause for semantic relations.
  - Example: *"'personal data' means any information relating to an identified or identifiable natural person ('data subject')**;** an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person**;**"* (GDPR)
- **Colon:** Mainly indicates multiple explanations following.
  - Example: *"'main establishment' means: (a) [...]; (b) [...];"* (GDPR)

Sentence's Root

Since we analyze each text segment for being a legal definition, we explore the root of the segment using dependency parsing. When the root is identified, the prototype investigates its POS tag as VERB (verb) or AUX (auxiliary). The detected root serves as a separator between definition and explanation, but only after a further check to prove that the processed text segment is actually a legal definition. After analyzing various legal acts, we concluded that most of the legal definitions used *"mean"* as a root, except for the rare occasions when *"include"* and *"be"* were mentioned. In order to generalize the approach, we examined the root's lemma for being a synonym of one of these three verbs.

Some examples of the roots in the legal definitions:

[9] `http://data.europa.eu/eli/reg/2018/1999/oj`, Definition 11

- **Mean**: *"'filing system' **means** any structured set of personal data [...]"* (GDPR)
- **Include**: *"'fishing trip' for a fishing vessel **includes** the time from its entry into until its departure [...]"*[10]
- **Be**: *"'length of optimal selectivity (Lopt)' **is** the average length of capture[...]"*[11]

Definitions Patterns

Based on [17], we summarized some of the definitions patterns that strongly impact correct extraction and annotation. Furthermore, we explain how the approach proceeds if it identifies one of the patterns.

From here on, we consider a definition as a legal term in apostrophes and an explanation as a meaning of the legal term.

- **One definition is part of another definition:** Save all definitions that contain another definition and check every mention of a shorter definition, whether it refers to one of the longest ones first.
  - Example: *personal data* and *personal data breach* (GDPR)
- **One definition to one explanation (1:1 relationship):** Save the definition and the explanation as (definition, explanation) pair.
  - Example: *"'filing system' means any structured set of personal data [...]"* (GDPR)
- **Multiplicity of definitions (M:1 or M:N relationship):** If commas, '*and*' or '*or*' are present in the definition part, including opening and closing apostrophes, each definition is split and saved separately with an explanation.
  - Example: *"'C1 tyres', 'C2 tyres' and 'C3 tyres' means tyres belonging to the respective classes set out in Article 8(1) of Regulation (EC) No 661/2009;"*[12]
- **Multiplicity of explanations (1:N or M:N relationship):** If the explanation part consists of more than one text block and includes a colon, then each explanation is saved together with the sentence's root separately.
  - Example: *"'main establishment' means: (a) [...]; (b) [...];"* (GDPR)

[10] `http://data.europa.eu/eli/reg/2019/833/oj`, Definition 25
[11] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 50
[12] `http://data.europa.eu/eli/reg/2020/740/oj`, Definition 1

Procedure

The process of identifying and extracting legal definitions in regulatory documents is presented in Algorithm 1.

Algorithm 1 Find and extract legal definitions

Require: *soup* (parse tree)
Ensure: *definitions_list*

```
function FIND_DEFINITIONS(soup)
    definitions_list ← list(tuple())
    start_class ← FIND_ARTICLE_WITH_DEFINITIONS(soup)
    end_class ← FIND_ARTICLE_FOLLOWING_DEFINITIONS(soup)
    for element in start_class.next_siblings do
        if element == end_class then
            break
        end if
        text ← element.text
        if "/" in text then
            nlp ← spacy.load("en_core_web_sm")
            doc ← nlp(text.split(";")[0])          ▷ consider only the first independent clause
            definition_set ← set()                                  ▷ for multiple definitions
            explanation_set ← set()                                 ▷ for multiple explanations
            first_verb ← None
            if token in doc then                                    ▷ searching for the sentence's root
                if token.dep_ == ROOT then
                    if token.pos_ == "VERB" or token.pos_ == "AUX" then
                        first_verb ← token
                    end if
                end if
            end if
            if first_verb in not None and is_synonym(first_verb.lemma_) then
                definition ← text[: first_verb.idx].strip()
                explanation ← text[first_verb.idx :].strip()
                d ← [s for s in definition.split("\n") if s != ""]
                e ← [s for s in explanation.split("\n") if s != ""]
                definition_set ← INVESTIGATE_DEFINITIONS(d)
                explanation_set ← INVESTIGATE_EXPLANATIONS(e)
                definitions_list ← SAVE_IN_LIST(definition_set, explanation_set)
            end if
        end if
    end for
    return definitions_list
end function
```

Find_article_with_definitions: searches for the article named *"Definitions"*.

Find_article_following_definitions: searches for the beginning of the article right after the one with definitions by applying regular expressions.

Investigate_definitions: checks whether multiple definitions occur by searching for a listing pattern, saves the found definition(s) in *definition_set*.

Investigate_explanations: checks whether multiple explanations occur by searching for ":" and controlling the length of *e*, saves the found formatted explanation(s) together with the first verb in *explanation_set*.

Save_in_list: saves each definition with each corresponding explanation.

While parsing the entered regulatory document, the prototype studies it for the article with legal definitions and iteratively examines each text segment until the end of the article (lines 3-8). If the text contains apostrophes, the tool cuts the text fragment by a semicolon and considers only the first clause of the possible explanation (line 12). Equivalent to [18], we are confronted with the issue that it is not clear where the explanation starts, so we assume that the sentence's root named *first_verb* (line 15) is the leading indicator of where the definition stops and the explanation begins. In order to find the sentence's root, tokenization is applied (line 16), and all tokens are investigated for being a root using dependency parsing (line 17). After the root is found, the token is examined for being a verb or auxiliary in line 18. If this condition holds true, the current token is assigned to the *first_verb*. In line 23, the prototype checks whether the sentence's root exists and if its lemma is a synonym of one of the most probable legal definitions verbs. Subsequently, the prototype separates the definition and explanation parts (lines 17-18) and splits both into text blocks, covering the case of multiplicity and removing the enumeration (lines 19-20). The text block may further contain multiple values, which must be split and stored apart; therefore, the *investigate* function explores the listing pattern and extracts single values, saving them into the set (lines 21-22). Since the approach mainly needs 1:1 relationships for the semantic relation analysis, we store each definition and each explanation as a pair (definition, explanation) using *save_in_list* function in line 23. Lastly, the algorithm returns a list of all the pairs (definition, explanation).

In the presented approach for definition extraction, we applied NLP techniques such as tokenization, lemmatization, dependency parsing, and POS tagging to examine the sentence's roots and decide about the belonging of sentences to the definitions. We refrained from using regular expressions as in [5], [16], [18], [23], although it could be an alternative solution for detecting hypothetical legal definitions.

Attaching Annotations

After extracting legal definitions from the regulatory document and storing them as a list of tuples (definition, explanation), the tool proceeds with investigating the whole document and attaching annotations containing explanations to the detected definitions, partially covering RQ3. Since users require all meanings for each legal term (1:N relationship) in order to interpret the legal text correctly, the approach further provides a dictionary where the keys refer to each definition, and values are the sets with corresponding explanations.

Idea

The process of attaching annotations is illustrated in Figure 3. Since the primary intention is to modify a regulatory document, the algorithm must provide a suitable structure for the parsing tree (Stage 1). After adjusting the tree, the approach extracts each text segment (Stage 2) and studies it for mentioning any legal definition (Stage 3). If the algorithm detects at least one term, it rewrites the text segment's content and replaces every legal definition with an annotation representing an explanation of the term (Stage 4).

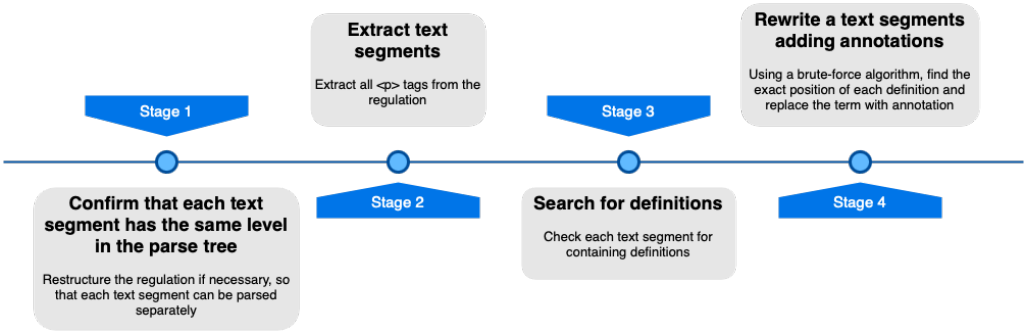


Figure 3
The four stages demonstrate the algorithm of assigning annotations containing explanations to each legal term occurring in the regulation.

Considering that text segments declare one main idea and every definition is mentioned explicitly in each sentence (Subchapter 3.1), the algorithm searches for the first mention of the legal definition and assigns annotations to it. If the text segment, which showed to be compact, contains the same definition in almost every sentence, the presence of the annotation by each mention can reduce visibility and readability. Moreover, in order to attach a correct annotation, each definition is tested

for the pattern *"One definition is part of another definition"*, as well as whether the annotation was already assigned to the corresponding position in the text.

Procedure

Algorithm 2 Adding annotations to the regulation

```
Require: soup (parse tree)
 1: function ADD_ANNOTATIONS(soup)
 2:     if soup.find("div", id = "001") is not None then
 3:         for div in soup.find_all("div") do
 4:             div.unwrap()
 5:         end for
 6:     end if
 7:     for sentence in soup.find_all("p") do
 8:         for (definition, explanation) in definitions_list do
 9:             if definition in sentence.text then
10:                 text ← sentence.text
11:                 sentence.clear()
12:                 def_positions ← sorted(ALL_DEF_IN_TEXT(text), key = lambda x : x[2])
13:                 start_index ← 0
14:                 for (d, e, start, end) in def_positions do
15:                     sentence.append(text[start_index : start])
16:                     tag ← CREATE_NEW_TAG(soup, text, d, e, start, end)
17:                     sentence.append(tag)
18:                     start_index ← end
19:                 end for
20:                 sentence.append(text[start_index :])
21:             end if
22:         end for
23:     end for
24: end function
```

Create_new_tag: creates a new tag with attributes *data_tooltip* and *style*.

Beginning with analyzing the tree structure of regulatory documents, we discovered that some acts use tag for each article and locate all text segments inside the tag. In contrast, others position the text segments on the same tree level. To generalize the approach, we restructure regulatory documents with and put all text segments on the same tree level (lines 2-6) and iterate over them (line 7). Each fragment is investigated for any occurring legal definition (line 9), and if at least one is discovered, the approach begins with a brute-force algorithm.

First, the approach caches the text segment's content (line 10) and removes it (line 11). Then, it calls the function *All_def_in_text*, which is presented in Algorithm 3. The function checks the text for definitions and possible occurrences of a longer definition. Then, depending on whether the position

is still available, it saves the obtained definition in a set together with its starting and ending position. Since the text segment can embrace multiple definitions, the algorithm sorts the returned set based on the start position (line 12) in order to recreate the sentence correctly. Subsequently, we partially reconstruct the segment, storing the part before the definition (line 15), creating an annotation on the position of the detected term (line 16), and adding it to the text segment (line 17). The end of the definition becomes the starting point, and the procedure repeats until all definitions receive an annotation (line 18). Lastly, the end of the segment is appended (line 20).

The example of the procedure is illustrated in Figure 4. The initial text segment is:

> *"This Regulation lays down rules relating to the protection of natural persons with regard to the processing of personal data and rules relating to the free movement of personal data."* (GDPR, Article 1, point 1)

Starting with removing the content of the text segment (phase 1), the algorithm calls *All_def_in_text*, which returns two first mentions of the terms *"processing"* and *"personal data"* and their positions in the text. After sorting the definitions, the approach reinstates text before *"processing"* (phase 2) and attaches an annotation with the term's explanation of the definition (phase 3), further concatenating it to the restored text. Then, the content before another legal definition *"personal data"* is attached (phase 4) and, equivalently to *"processing"*, the annotation is assigned to *"personal data"* (phase 5). Finally, the algorithm re-establishes the text following the obtained terms (phase 6).

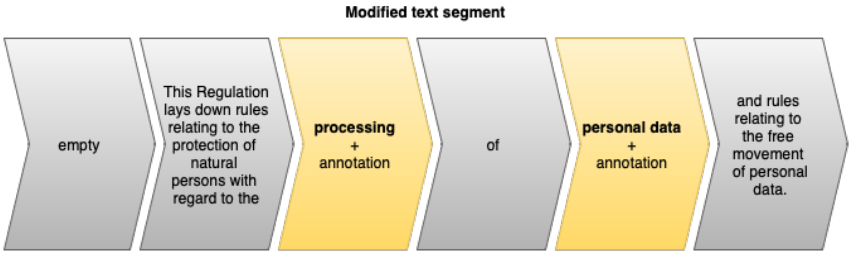


Figure 4
Partial reconstruction of the text segment, where initially the text before the first occurring legal term is restored, then the annotation is assigned to the definition, concatenating it to the existing string. The procedure repeats until the last diverse legal term is processed. Lastly, the algorithm connects the rest of the segment.

Algorithm 3 Detecting various legal definitions in the text segment

Require: *text*
Ensure: *definitions_in_text*

```
function ALL_DEF_IN_TEXT(text)
    definitions_in_text ← set(tuple())
    starts_and_ends ← set(tuple())
    for definition, explanation in definitions_dictionary.items() do
        if definition in text then
            other_definitions ← SUB_DEFINITIONS(definition)
            if len(other_definitions) != 0 then          ▷ if definition is in other definition(s)
                for d in other_definitions do
                    if d in text then
                        match ← re.search(d, text)       ▷ searches for the first mention of d
                        if match is not None then
                            s, e ← match.start(), match.end()
                            if not POSITION_TAKEN(start_and_end, s, e) then
                                definitions_in_text.add(d, definitions_dictionary[d], s, e)
                                starts_and_ends.add(s, e)
                            end if
                        end if
                    end if
                end for
            end if
            match ← re.search(definition, text)
            if match is not None then
                start, end ← match.start(), match.end()
                if not POSITION_TAKEN(start_and_end, start, end) then
                    definitions_in_text.add(definition, value, start, end)
                    starts_and_ends.add(start, end)
                end if
            end if
        end if
    end for
    return definitions_in_text
end function
```

Sub_definitions: returns a set with other definitions which contain the definition inside.

Position_taken: checks if another definition was already saved on this position.

Extracting Sentences Including Definitions

Due to efficiency reasons, the sentence extraction is performed in parallel to the attachment of annotations, and all detected text segments are stored as a sentence for a corresponding definition. Since legal terms are always mentioned explicitly (as illustrated in Figure 5[13]), and we extract the whole text segment describing one idea, the algorithm does not have to examine the sentences for being connected and referring to the same subject, which is not named directly. Additionally, the

[13] http://data.europa.eu/eli/reg/2019/517/oj

algorithm investigates each text segment for all legal definitions without splitting it, providing that every legal term is covered (Fig. 5, point 4).

SECTION 2

Registry

Article 8

Designation of the Registry

1. The Commission shall adopt delegated acts in accordance with Article 18 to supplement this Regulation by establishing the eligibility and selection criteria and the procedure for the designation of the Registry.

2. The Commission shall set out the principles to be included in the contract between the Commission and the Registry, by means of implementing acts. Those implementing acts shall be adopted in accordance with the examination procedure referred to in Article 17(2).

3. The Commission shall designate an entity as the Registry following the completion of the procedure referred to in paragraphs 1 and 2.

4. The Commission shall enter into a contract with the designated Registry. The contract shall specify the rules, policies and procedures for the provision of services by the Registry and the conditions according to which the Commission is to supervise the organisation, administration and management of the .eu TLD by the Registry. The contract shall be limited in time and shall be renewable once without the need to organise a new selection procedure. The contract shall reflect the obligations of the Registry and shall include the principles and procedures on the functioning of the .eu TLD laid down on the basis of Articles 10 and 11.

5. By way of derogation from paragraphs 1, 2 and 3, the Commission may, where imperative grounds of urgency exist, designate the Registry by means of immediately applicable implementing acts in accordance with the procedure referred to in Article 17(3).

Figure 5
Example of a regulation where the whole section refers to a single legal definition. Nonetheless, each sentence explicitly mentions a term and does not apply any indirect references as the pronoun "it".

Nevertheless, as established in [23], it is essential to differentiate between two types of text segments that appear in legal text: sentences and lists. The sentences are independent text blocks (Fig. 6, Article 9), while in lists, the first sentence forms a correct sentence with each of the separate list items (Fig. 6, Article 10). In [23], the authors assumed that deriving and classifying each of those items would solve some issues. Therefore, the algorithm does not extract the lists as a complete block compared to sentences but investigates each point separately.

Article 9

Characteristics of the Registry

1. The Registry shall be a not-for-profit organisation. It shall have its registered office, central administration and principal place of business within the territory of the Union.

2. The Registry may impose fees. Those fees shall be directly related to the costs incurred.

Article 10

Obligations of the Registry

The Registry shall be required to:

(a) promote the .eu TLD across the Union and in third countries;

(b) comply with the rules, policies and procedures laid down in this Regulation, with the contract referred to in Article 8(4), and, in particular, with Union data protection law;

Figure 6
Difference between sentences and lists used in articles of regulatory documents. The sentences use a numbering format for enumerating the referring points, and each item is entirely independent. In contrast, lists apply an alphabetical format, and all the items refer to the text fragment right before the beginning of the list.

In addition to the extracted text segments, a returned text file contains the list of all articles, where each legal definition was mentioned, so the users interested in acquiring further information regarding one of the terms can effortlessly find its location in the lengthy regulation.

Relation Extraction

The final phase of the approach consists of resolving semantic relations among definitions such as hyponymy, meronymy, and synonymy. The classification of semantic relations between definitions within the regulation is required for various semantic interpreting tasks, such as textual entailment and inquiry answering. However, in most circumstances, identifying a semantic relation between terms is rather challenging [9]. Figure 7 demonstrates the whole relation extraction process. To simplify the recognition of semantic relations among definitions, the approach manages the explanation of each definition (stage 1). It examines it for matching one of the patterns of the relations, summarized below (stage 2). Based on the observed pattern, the explanation is split if required. By applying NLP techniques like dependency parsing, tokenization, and POS tagging, the algorithm extracts at least one noun phrase chunk (stage 3), removes stop words, and stores it as a semantic relation (stage 4). By default, the approach assumes that the first identified noun phrase chunk is a hyperonym of the term. However, rarely can some exceptions be found where the definition has no semantic relations and only explains the concept behind the definition (e.g., *"'produced from GMOs' means derived in whole or in part from GMOs but not containing or consisting of GMOs;"*[14]).

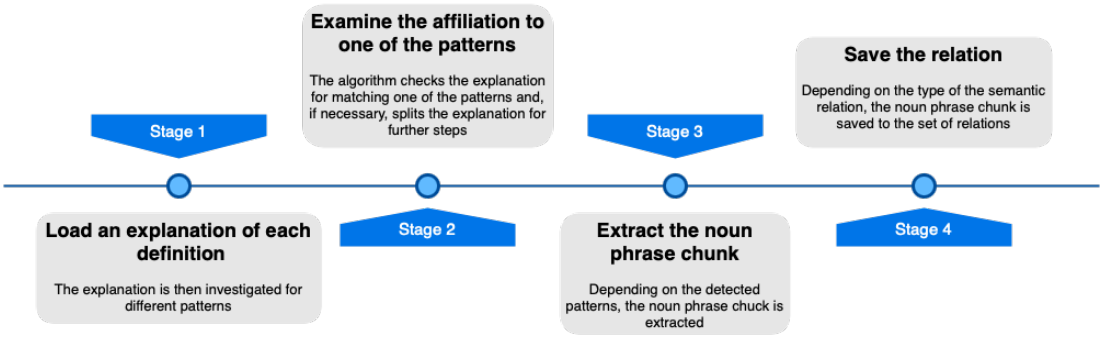


Figure 7
Four stages of relation extraction where the algorithm analyzes each explanation based on pattern matching and extraction of noun phrase chunks.

[14] `http://data.europa.eu/eli/reg/2018/848/oj`, Definition 59

Relations Patterns

This section lists all considered patterns of possible semantic relations and how the prototype processes them. If none of the patterns is detected, we assume the hyponymy, where the first identified noun phrase chunk is a hyperonym.

- **Explanation contains *"for"* and a colon:** The prototype analyzes the part of the sentence following the colon.
  - Example: *"'mesh size' means: (i)* ***for*** *knotted netting****:*** *the longest distance [...]; (ii)* ***for*** *knotless netting****:*** *the inside distance [...]"*[15]
- **Explanation contains a colon and *"in the context of"*, *"for"*, or *"as regards"*:** The prototype analyzes the part of the sentence, following this pattern and comma.
  - Example: *"'main establishment' means: (a)* ***as regards*** *a controller with establishments in more than one Member State****,*** *the place [...]; (b)* ***as regards*** *a processor with establishments in more than one Member State****,*** *the place [...]"* (GDPR)
  - Example: *"'good status' means****:*** *(a)* ***for*** *surface water, [...]; (b)* ***for*** *groundwater, [...]"*[16]
  - Example: *"'early efforts' means****:*** *(a)* ***in the context of*** *the assessment of a potential gap between the Union's 2030 target for energy from renewable sources and the collective contributions of Member States, [...]; (b)* ***in the context of*** *Commission recommendations based on the assessment pursuant to point (b) of Article 29(1) with regard to energy from renewable sources, [...]"*[17]
- **Explanation's first noun phrase chunk is *"following"*:** Proceed with the next noun phrase chunk.
  - Example: *"'project promoter' means one of the following: (a) [...]; (b) [...];"*[18]
- **Explanation contains multi-word noun phrase chunks:** Similar as in [26] if a noun-phrase chuck contains a part of a multi-word explanation, the chunk is expanded to contain the complete name.
  - Example: *"'regional indicative programme' means a multi****-****country indicative programme [...]"*[19]

[15] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 34
[16] `http://data.europa.eu/eli/reg/2020/852/oj`, Definition 22
[17] `http://data.europa.eu/eli/reg/2018/1999/oj`, Definition 20
[18] `http://data.europa.eu/eli/reg/2022/869/oj`, Definition 8
[19] `http://data.europa.eu/eli/reg/2021/947/oj`, Definition 3

- **Explanation contains enumeration (comma, *"and"*, *"or"*) following the first noun-phrase chunk:** All the subsequent noun-phrase chunks in the pattern are stored as hyperonyms.
  - Example: *"'third party' means a natural or legal person**,** public authority**,** agency **or** body other than the data subject, controller, processor and persons who, under the direct authority of the controller or processor, are authorised to process personal data;"* (GDPR)
- **Explanation is defined in another legal document:** Process as a hyponymy and store a definition as a hyperonym of itself.
  - Example: *"'financial product' means a financial product as defined in point (12) of Article 2 of Regulation (EU) 2019/2088;"*[20]
- **Explanation contains *"type of"*, *"kind of"*, or *"form of"*:** Process as a hyponymy.
  - Example: *"'static nets' means any **type of** gillnet[...]"*[21]
- **Explanation contains *"part of"*, *"piece of"*, or *"portion of"*:** Process as a meronymy.
  - Example: *"'codend' means the rearmost **part of** the trawl[...]"*[22]
- **Different definitions have the same explanation:** Process as a synonymy.
  - Example: *"'purse seine' or 'ring nets' means any surrounding net [...]"*[23]

Since the literature revision depicted that pattern matching can be a suitable strategy [19], [23], [24], [26], but it directly depends on mentioning all existing patterns [5], [17], [18], other relation patterns can be investigated in future work and attached to the approach.

Hyponym-Hyperonym Relation

According to Cruse [30], of all the semantic relations, hyponymy appears across the broadest range of grammatical categories and domains. Based on [30], [31], we define hyponymy as following:

Definition 3.5.1 (Hyponymy) Hyponymy, also known as is-a, a-kind-of, taxonomic, superordinate -subordinate, genus-species and class-subclass relation, is the cognitive processing of information and an essential means of classifying words, where **hyponym** refers to the narrower term/concept and **hyperonym** is the more general term/concept.

[20] `http://data.europa.eu/eli/reg/2020/852/oj`, Definition 3
[21] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 23
[22] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 33
[23] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 21

This semantic relation indicates inclusion and is the most common type of relation among legal definitions. As a result, the approach initially presumes hyponymy, where the definition belongs to a hyponym and the first occurring noun phrase chunk to a hyperonym [26]. As an example, we consider Definition 1 from GDPR, where *"personal data"* is a hyponym and *"information"* is a hyperonym:

> *"'personal data' means any information relating to an identified or identifiable natural person ('data subject'); [...]"*

In this case, all instances of *personal data* are *information*, the set of *personal data* instances is a subset of *information*, and the meaning of *personal data* is included in the meaning of *information* [30].

The legal definitions containing *"type of"*, *"kind of"*, and similar also specify hyponymy. In this particular circumstance, the approach proceeds with searching for the noun phrase chunk and stores the second match as a hyperonym.

As mentioned in the pattern *"Explanation is defined in another legal document"*, some regulations contain definitions which point to another regulation not included in the corpus. In order to deal with this issue, Ferneda et al. [1] suggested two possible actions: either find and retrieve the stated regulation from an external source or neglect the orphan definitions. Since the first option is costly, the approach handles these definitions as hyponymy and, therefore, points to users the necessity to look up. Examples of this kind of definition are:

> *"'surface water' means surface water as defined in point 1 of Article 2 of Directive 2000/60/EC;"*[24]
> *"'support scheme' means support scheme as defined in point (5) of Article 2 of Directive (EU) 2018/2001;"* [25]

Meronym-Holonym Relation

Compared to hyponymy, which can exist within concepts, meronymy relations are between concepts [31].

[24] `http://data.europa.eu/eli/reg/2020/852/oj`, Definition 19
[25] `http://data.europa.eu/eli/reg/2018/1999/oj`, Definition 24

Definition 3.5.2 (Meronymy) Meronymy, also known as part-whole relation and partonymy, refers to the relation between a concept/entity and its constituent parts, where **melonym** belongs to the part of the term/concept and **holonym** to the whole term/concept [31].

Among legal definitions, meronymy is infrequent in contrast to hyponymy. The algorithm identifies meronymy if the first detected noun phrase chunk points to the expressions *"part of"*, *"piece of"*, or similar. Then, the approach memorizes the type of relation and searches for the next noun phrase chunk to save it as a holonym. Examples of meronymy:

> *"'veranda' means an additional, roofed, uninsulated, outdoor part of a building [...]"*[26], where *veranda* is a meronym and *building* is a holonym.
> *"'the Regulatory Area' means that part of the Convention Area [...]"* [27], where *the Regulatory Area* is a meronym and *Convention Area* is a holonym.

Synonymy

The algorithm further covers synonymy, which is present if at least two legal definitions have an equal explanation.

Definition 3.5.3 (Synonymy) Synonymy is a semantic relationship where terms consider synonymous in case they have identical meanings.

In order to identify synonymy, no NLP technique is required. The approach simply collects the definitions with the same explanation and stores all the detected terms together as synonyms. Examples of synonymy:

> *"'Danish seine' or 'Scottish seine' means an encircling and towed gear[...]"* [28], where *Danish seine* and *Scottish seine* are synonyms.
> *"'immersion time' or 'soak time' means the period from the point of time [...]"*[29], where *immersion time* and *soak time* are synonyms.

[26] `http://data.europa.eu/eli/reg/2018/848/oj`, Definition 28
[27] `http://data.europa.eu/eli/reg/2019/833/oj`, Definition 3
[28] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 18
[29] `http://data.europa.eu/eli/reg/2019/1241/oj`, Definition 41

Procedure

Algorithm 4 Identifying semantic relations

```
Require: definitions                                              ▷ list of definitions
function IDENTIFY_RELATIONS(definitions)
    nlp ← spacy.load("en_core_web_sm")
    for (definition, explanation) in definitions do
        if IS_IN_ANOTHER_DOCUMENT(definition, explanation) then
            SAVE_TO_HYPONYMY(definition, definition)
        end if
        sentence ← PREPARE_EXPLANATION(explanation)
        doc ← nlp(sentence)
        multi_word, meronymy ← False, False
        for noun in doc.noun_chunks do
            if meronymy then
                SAVE_TO_MERONYMY(noun, definition)
                break
            end if
            if noun.root.text == "following" or IS_HYPONYMY(noun.root.text) then
                continue
            end if
            if IS_MERONYMY(noun.root.text) then
                meronymy ← True
                continue
            end if
            if SINGLE_RELATION(doc[noun.end:]) and not multi_word then
                SAVE_TO_HYPONYMY(noun, definition)
                break
            end if
            if multi_word then
                PROCESS_MULTIWORD(doc)
                break
            end if
            SAVE_TO_HYPONYMY(noun, definition)
        end for
    end for
    FIND_SYNONYMS( )
end function
```

Is_in_another_document: indicates whether the pattern *"Explanation is defined in another legal document"* is detected.

Prepare_explanation: returns a sentence, checking for the patterns like *"Explanation contains "for" and a colon"* or *"Explanation contains a colon and "in the context of", "for", or "as regards""* and cutting the sentence accordingly.

Save_to_meronymy: after removing stop words, saves the noun phrase chunk to the *relations* set as a holonym.

Is_hyponym: indicates whether the pattern *"Explanation contains "type of", "kind of", or "form of""* is detected.

Is_meronymy: indicates whether the pattern *Explanation contains "part of", "piece of", or "portion of""* is detected.

Save_to_hyponymy: after removing stop words, saves the noun phrase chunk to the *relations* set as a hyperonym.

Single_relation: examines the pattern *"Explanation contains enumeration (comma, "and", "or") following the first noun-phrase chunk"* and checks for multi-word noun phrase chunks, setting *multi_word* variable true and returning false.

Process_multiword: concatenates the current processed noun phrase chunk and the following chunk, saving the resulting string in the *relations* set.

Find_synonyms: searches for all definitions with the same explanation and adds them together to the *relations* set as synonyms.

The whole algorithm for relation extraction is based on pattern matching and works with noun phrase chunks as in [26]. Similar NLP techniques as described in [9], [26], [27] were applied with the purpose of deriving semantic relations related to legal definitions. The algorithm iterates over each definition and its explanation (line 3) and searches for a specific pattern in order to identify the relation correctly. Starting with the *"Explanation is defined in another legal document"* pattern (lines 4-6), we examine an explanation for containing a definition inside of it and any pointing on another legal document, for example, with the expression *"as defined in point"*. This check is implemented by calling the *Is_in_another_document* function, and if the pattern is confirmed, the algorithm saves the relation as hyponymy. Subsequently, the algorithm investigates the explanation for further patterns such as *"Explanation contains "for" and a colon"* and *"Explanation contains a colon and "in the context of", "for", or "as regards""* by calling the *Prepare_explanation* function and splits the text accordingly to focus on the main clause (line 7). The resulting explanation is a final sentence on which the NLP techniques are applied (line 8). For the patterns *"Explanation contains multi-word noun phrase chunks"* and *"Explanation contains "part of", "piece of", or "portion of""* additional variables named *multi_word* and *meronymy* are created (line 9).

The iteration over noun phrase chunks begins from line 10. In case the *meronymy* variable is set to true, the algorithm saves the processing chunk as meronymy, removing stop words beforehand (lines 11-14). If the first noun phrase chunk refers to *"following"* or its root points to *"type"/"kind"*

together with the next token being *"of"*, the approach ignores the phrase and proceeds with the next chunk (lines 15-17). Following, the algorithm examines a chunk for indicating meronymy by examining its root again for *"piece"/"part"/"portion"* and the subsequent token for *"of"*, setting the variable *meronymy* to true if the pattern was identified and continuing the iteration (lines 18-21). The investigation of the following token is important since the indicating words can be simple hyperonyms (e.g., *"'equivalent tyre type' means a tyre type which is placed on the market [...]"*[30]).

The default case is covered in lines 22 to 25, where the algorithm identifies no patterns and, therefore, stores the detected noun phrase chunk as a hyperonym of the legal definition. Furthermore, the *Single_relation* function also examines the following token for referring to a multi-word. It sets the variable multi-word to true if the subsequent token is a dash and returns a false value, so the *Process_multiword* function can concatenate the current noun phrase chunk and the following one with the dash in between (lines 26-29). At the end of the for-loop, the algorithm assumes the *"Explanation contains enumeration (comma, "and", "or") following the first noun-phrase chunk"* pattern, stores the current noun phrase chunk as a hyperonym and proceeds with the iteration (line 30). Lastly, the *"Different definitions have the same explanation"* pattern is examined by calling *Find_synonyms* function (line 33).

The algorithm stores all the detected semantic relations in the global *relations* set, further collecting them in a file for users to download. In order to improve the understanding of how the legal terms are related, the document additionally illustrates a semantic hyponymy tree at the end of each file.

Visualization

After extracting the applicable data, using European legal information can still be difficult for users. Accessing relevant documents does not necessarily benefit fulfilling the information needs since operating regulatory documents is challenging without sufficient knowledge of the related legal system [32]. The secondary purpose of the approach is to offer a consistent user interface that demonstrates the retrieved legal information represented by legal definitions and their semantic relations by providing separate files with obtained data.

By appropriately visualizing the extracted legal definitions and semantic relations among them in a graphical interface, the purpose is to help European users cope with the complexity of regulatory

[30] `http://data.europa.eu/eli/reg/2020/740/oj`, Definition 24

documents' specific language, obtaining relevant information in a shorter time and correctly interpreting the legal text. Furthermore, the detected semantic relations encourage the users to make further inferences by investigating how definitions are semantically related. The approach returns the files visualizing the extracted information as follows:

- **File with legal definitions:** presents all extracted legal definitions and their explanations.
- **File with text segments:** lists all text segments mentioning each legal definition and the articles indicating their location, along with the number of hits.
- **File with semantic relations:** contains all identified semantic relations and the hyponymy tree for a better understanding of the levels of relationships.
- **File with annotated regulation:** introduces a whole regulation with annotations to legal definitions, consisting of their explanations for a more accurate interpretation of the legal text.

Addressing RQ4, various additional information that can be helpful for users can be extracted from the regulations, such as:

- **Full title:** indicates the primary subject matter and makes it possible to determine who is affected by the regulatory document [29].
- **Date of execution**
- **Number of legal definitions:** can point out the document's complexity and specificity.
- **The most frequent definitions**: indicates the importance of legal definitions to the intention and objective of the regulatory document.
- **List of articles where each legal definition occurs**: simplifies the search for a specific definition in the whole regulatory document.

As the focus of the approach lies in extracting legal definitions, one of the valuable factors that the prototype should display is the frequency of definitions in the text segments instead of the sentences. The reason is that the whole text segment can contain one definition, which does not automatically make it one of the most frequent legal definitions. The frequency is an important aspect of regulatory documents since it can indicate that the legal definition is essential to the meaning and objective of the legal document and highlights where the document places the emphasis.

Implementation

This paper concentrates on a practical contribution, specifically, implementing a prototype presented in Chapter 3 capable of extracting legal definitions from the Regulations of the European Parliament and of the Council provided by EUR-Lex and identifying semantic relations among them. The main contribution of this bachelor thesis includes an opportunity for the user to enter a CELEX number of a regulation as an input, and the tool is supposed to return a list of legal definitions as a text file, a list with text segments containing legal definitions as a text file, an annotated regulation as an HTML file, and a list of semantic relations as a text file. The source code repository of the approach is publicly accessible at `https://github.com/AnastasiyaDmrsk/Identification-and-Visualization-of-Legal-Definitions-and-Relations`.

The major technologies applied for the prototype are following:

Backend: Python 3, Django[31] web framework, BeautifulSoup[32] library for extracting data from HTML files, spaCy[33] library for NLP, WordNet[34], NLTK[35]

Frontend: HTML, CSS, Bootstrap[36]

In the coming sections, we first explain how the prototype verifies the CELEX number and regulations considered as an input in Section 4.1. Then we describe the implementation details of each phase addressed by the research questions, starting with explaining the usage of spaCy, as in [24], in order to identify legal terms and obtain them in Section 4.2. Next, in Section 4.3, we present the techniques used for extracting text segments containing definitions and assigning annotations to them to highlight relevant terms in the legal text. Then, we explain the exact approach for establishing semantic relations such as hyponymy, meronymy and synonymy in Section 4.4, again applying spaCy. Finally, in Section 4.5, we illustrate the prototype's layout and provide informative statistics based on the revealed information.

[31] `https://www.djangoproject.com/`
[32] `https://beautiful-soup-4.readthedocs.io/en/latest/`
[33] `https://spacy.io/`
[34] `https://wordnet.princeton.edu/`
[35] `https://www.nltk.org/`
[36] `https://getbootstrap.com/`

Input Verification

Derived from the findings mentioned in 3.1, the tool validates the entered CELEX number and the corresponding regulation before starting with the processing phase. Most of the attention is paid to checking the first and the third units of CELEX number, referring to a sector and a document type, as well as the length of the inputted number. For the Regulations of the European Parliament and of the Council, the sector must be '3' and the document type 'R'(e.g., **3**2016**R**0679 (GDPR)), along with the length equal to ten characters. In case the number passes the criteria, the prototype loads an URL *'https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:'* + *CELEX number* + *'&from=EN'* redirecting to EUR-Lex legal acts in the English Language. While the URL is determined as valid, it is also plausible that no regulation with this CELEX number is present. In this case, EUR-Lex returns a title *"The requested document does not exist"*, which the prototype catches and subsequently raises a validation error. Furthermore, the form studies whether the document contains an article called *"Definitions"* using the BeautifulSoup *find* function, and if not, informs the user that handling the submitted regulation is not possible due to the lack of legal definitions section.

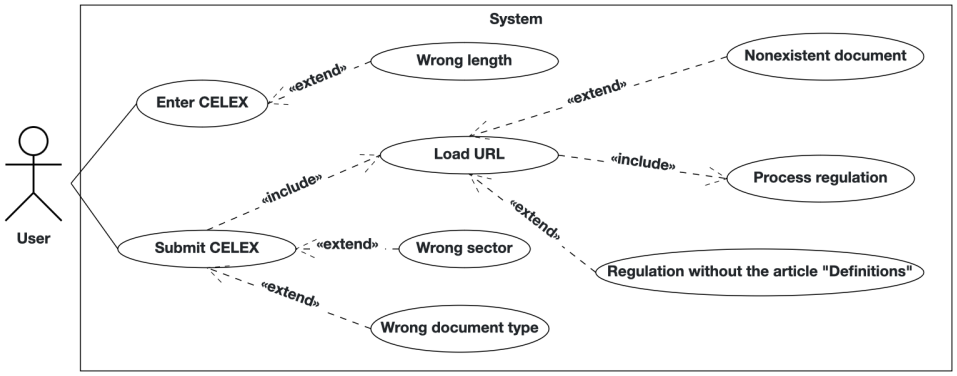


Figure 1
Use case diagram depicting how the user interacts with the system while entering the CELEX number.

Extracting Legal Definitions Using spaCy

After loading the valid regulatory document, the tool pursues the next step presented in 3.2 and extracts legal terms using the BeautifulSoup HTML parser. After detecting a hypothetical definition, the tool has to examine a sentence's root for being a synonym of the most common verbs pointing to a definition. Therefore, we integrate a lexical database WordNet applying NLTK since SpaCy has no integration of it. In case of a match, the prototype treats the obtained segment as a legal definition

and splits it into the definition and explanation parts, further storing the extracted definition in the dictionary as a 1:1 relationship to output a text file with all legal terms more efficiently. Then, as illustrated in Figure 2, each part iteratively investigates the M:N relationship and saves the spitted results in the according sets. Afterwards, the pairs *(definition, explanation)* are stored as a list of tuples for identifying semantic relations in the later phase and as a dictionary depicting the 1:N relationship for assigning annotations, including all possible explanations of each legal term.

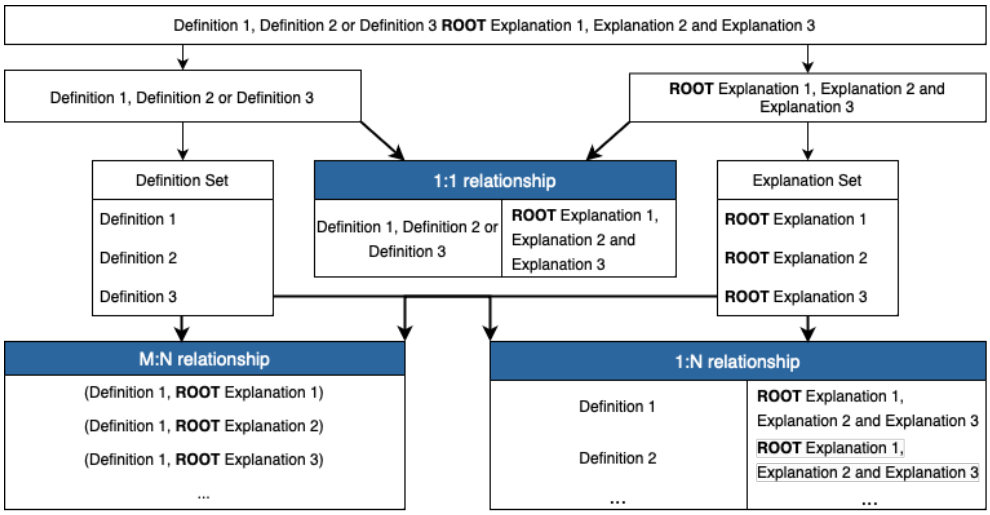
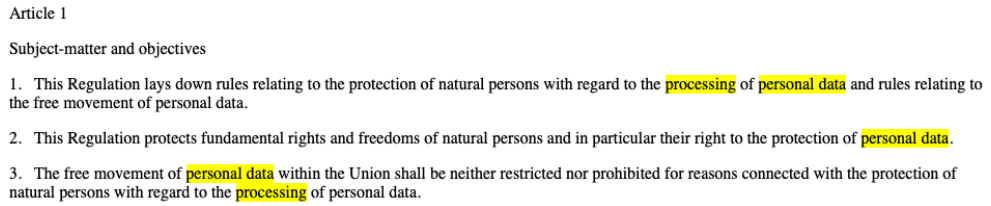


Figure 2
Three ways of storing legal definitions: a 1:1 relationship for outputting only legal definitions as a dictionary, a 1:N relationship for annotations as a dictionary, and a M:N relationship for relation extraction as a list of tuples.

Extracting Sentences and Attaching Annotations Using BeautifulSoup

Attaching annotations to the regulation and extracting sentences with definitions is implemented mutually with the support of BeautifulSoup, which extracts text segments marked with the tag <*p*> by calling the *find_all* function. Before starting with extraction, the prototype first examines the HTML structure of the legal document and the presence of the tag <*div*> with an *id* = *001*, which points to the first article. If the tag is detected, then all <*div*> tags are removed from the regulation using *unwrap()*, but their inner <*p*> tags shift to the same level of the parse tree. This procedure is necessary since if the tags do not belong to the same level in the parse tree, text segments are combined and represent a massive block of text. Then the prototype examines fragments for the occurring definitions and returns a set of tuples with the first mentions of the definitions *(definition, explanation, start_index, end_index)*. As illustrated in 3.3, we first sort the returned set depending on the *start_index* and then replace the definitions in the original content with the string values of a

new tag, further appending the derived string to the initial page element. The new tag represents a span with two attributes: *style = "background-color: yellow;"* and *data-tooltip = "definition + ' ' + explanation"*, where the explanation is a concatenation of all explanations of the corresponding term taken from the dictionary for 1:N relationships. The figures 3 and 4 display the resulting regulation by the example of GDPR, including annotations with and without hover. Lastly, the prototype stores the detected text segments in the sentences dictionary with a key equal to the occurring legal definition and the value representing a set of all text segments containing the definition.



Article 1

Subject-matter and objectives

1. This Regulation lays down rules relating to the protection of natural persons with regard to the processing of personal data and rules relating to the free movement of personal data.

2. This Regulation protects fundamental rights and freedoms of natural persons and in particular their right to the protection of personal data.

3. The free movement of personal data within the Union shall be neither restricted nor prohibited for reasons connected with the protection of natural persons with regard to the processing of personal data.

Figure 3
Emphasizing annotations in Article 1 of GDPR by highlighting the first mentions of legal definitions in each text segment.

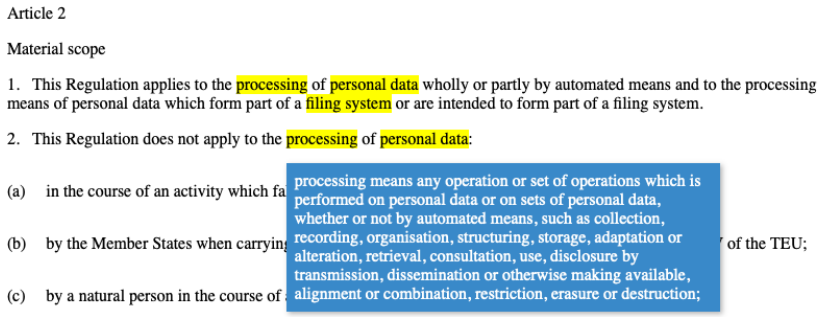


Figure 4
Hovering over the annotation attached to a legal definition in Article 2 of GDPR provides a window with an explanation of the term.

Identifying Semantic Relations Using spaCy

To obtain hyponymy and meronymy, we apply spaCy's function called *noun_chucks()* and iterate over each explanation examining the noun phrase chunks for the relations patterns. Depending on the resulting relation, the detected chunk is saved as follows:

- **Hyponymy:** *"definition + ' is a hyponym of ' + hyperonym"*
- **Meronymy:** *"definition + ' is a meronym of ' + holonym"*

To provide a better overview, the semantic tree for hyponymy is displayed at the end of the text file, where the roots are the hyperonyms, which are not simultaneously part of any hyponyms' set (Fig. 5, GDPR).

```
set
| processing
| | pseudonymisation
| | cross-border processing
structured set
| filing system
breach
| personal data breach
agency
| controller
| | main establishment
| third party
| processor
| | main establishment
| recipient
organisation
| international organisation
service
| information society service
```

Figure 5
Part of the hyponymy semantic tree refers to legal definitions extracted from GDPR, which further depicts multiple levels of hyponymy.

Since we take the M:N relationships into account, where many definitions can have many explanations, we investigate the definitions with the same explanation from the dictionary of legal definitions used previously in attaching annotations and save them into the list of tuples, where each tuple contains synonymous terms. These definitions are then added to the relations text file in the form *"definition 1 + ', ' + ... + ', ' + definition n + ' are synonyms'"*. To demonstrate it, the segment of the resulting relations' text file including synonyms after processing the Regulation on the labelling of tyres with respect to fuel efficiency and other parameters[37] is illustrated in Fig. 6. For the semantic tree, no changes are needed since the synonyms have the same hyperonyms and are placed together as child nodes.

```
C1 tyres is a hyponym of tyres
C1 tyres, C2 tyres, C3 tyres are synonyms
C2 tyres is a hyponym of tyres
C3 tyres is a hyponym of tyres
```

Figure 6
Example of illustrating synonyms in the resulting relations' file.

[37] http://data.europa.eu/eli/reg/2020/740/oj

The resulting text file depicting the obtained semantic relations is sorted alphabetically to simplify the search for the specific legal definition. Afterwards, the document presents a hyponymy tree, so the user can effortlessly find terms with the same hyperonym and potentially gain a more profound understanding of their meaning.

Visualization

Beginning with the layout of the prototype, we applied Django Forms and Bootstrap to design the home page demonstrated in Figure 7 for users to submit a requested CELEX number. In case of a validation error, the users receive an error message based on the occurred problem, while the tool stops processing the entered number.

Figure 7
Start page of the prototype, where users can enter a CELEX number of a regulation that is then directly verified after pressing an upload button.

If the regulation passes the criteria, the prototype handles the document and redirects users to the resulting page illustrated in Figure 8. With the purpose of increasing usability, the tool extracts the full title of the entered regulation. Additionally, it displays five options: the user can be redirected to the original source in EUR-Lex or download all the generated output files in the specified format.

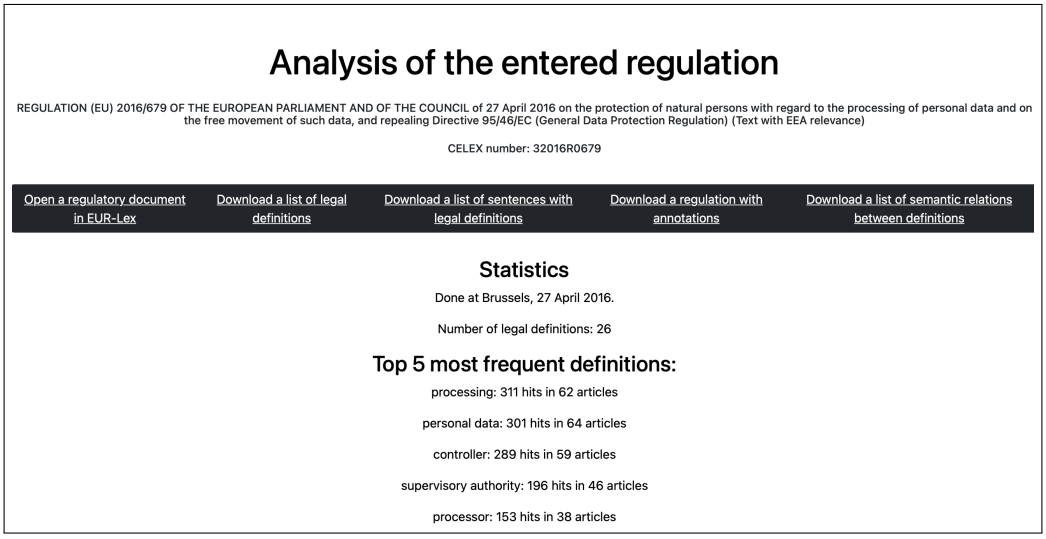


Figure 8
Result page of the prototype after processing the regulation. The full title is displayed, along with the CELEX number, at the very top of the page. Following this, the web service provides five buttons to redirect or download the extracted information. In the end, the statistics concerning legal definitions are presented.

Furthermore, the prototype renders the statistics relying on the regulation, such as the date of execution, a critical detail in regulatory documents concerning document validity. The number of legal definitions can also be a relevant factor that indicates the complexity of the regulation and whether it covers a wide range of areas or circumstances since each legal term attaches more details and specificity to the legal document, refining and compounding it at the same time. Since the focus of this thesis lies in extracting and examining legal definitions, the prototype further provides a deeper look into the usage of every one of them. It obtains the most frequent terms based on the number of text segments referring to them, also mentioning the number of articles where they occur. To gain a better overview of which articles a specific definition appears in, a user can download a text file with text sentences where each term lists all referring articles.

Evaluation

A prototypical implementation of the research methodology mentioned in Chapter 3 is provided and applied for the evaluation. The evaluation's emphasis is to determine whether the approach is capable of detecting all legal definitions and their explanations in legal text, correctly annotating and extracting text segments mentioning them, and identifying semantic relations among them by various Regulations of the European Parliament and of the Council published on EUR-Lex, which include an article referring to legal definitions. Moreover, the visualization of the extracted information addressed in RQ3 should meet users' needs in such aspects as usability and readability, as well as the interface of the tool should be easy to understand and utilize.

In order to analyze the results and success of the prototype in legal information retrieval, we devise several questions, which are further addressed in Section 5.2:

Q1 *Were all legal definitions extracted? Were definitions extracted correctly?*

Q2 *Were all text segments mentioning legal definitions annotated and extracted? Were text segments discovered and annotated correctly?*

Q3 *Were semantic relations among definitions correctly identified, i.e., how precise is the approach?*

The first question refers to RQ1 from Section 1.2 and studies whether the tool identified all definitions fully and correctly in the legal text, especially emphasizing the identification of the pattern *"One definition is part of another definition"*. Furthermore, Q2 points to RQ3 since it examines the fullness and correctness of annotations and a returned set with text segments containing legal terms. For the modified regulation, it is necessary to control whether the text segments were reconstructed correctly by applying a brute-force algorithm, as well as whether legal terms were placed in the correct order with their correct explanations. Meanwhile, Q3 relates to RQ2, which investigates how precisely the prototype established semantic relations among legal definitions like hyponymy, meronymy, and synonymy. Thereby, the core tasks of the approach are evaluated separately to facilitate debugging in case of an error. The visualization mentioned in RQ3 and RQ4 is further assessed with the functional and non-functional requirements in Section 5.3.

Data Set

To evaluate the approach properly, we consider 17 Regulations of the European Parliament and of the Council and GDPR to estimate the correctness of the approach in extracting legal definitions and identifying semantic relations. The selection criteria were to reject the regulations which had no article called *"Definitions"* since the prototype declines this type of document in the early stage as demonstrated in 5.2. The regulations were selected randomly in March 2023 by searching for the keyword *"regulation*" and then sorting the results by relevance on EUR-Lex and investigating the first 20 hits for the articles with definitions. Table 1 lists all picked regulations by CELEX number. To better depict the variety of the chosen legal documents, we summarized them by topics provided by EUR-Lex[38] in Figure 1.

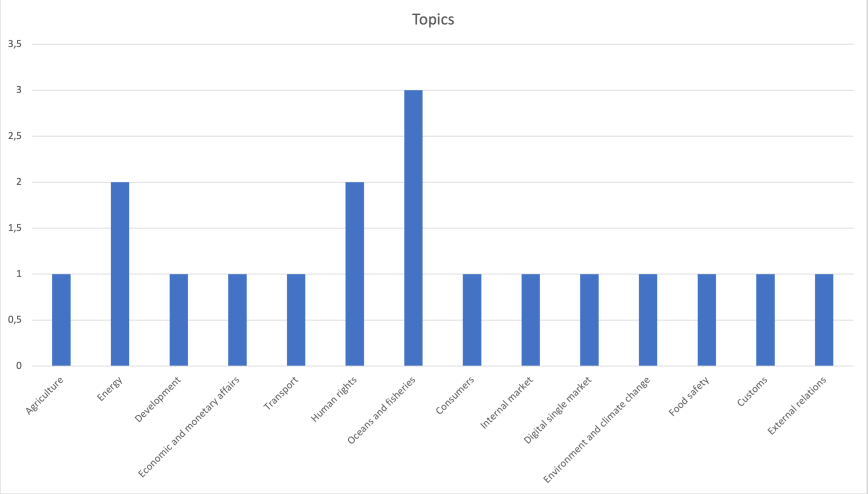


Figure 1
Distribution of selected regulations' topics for evaluation, overall 14 subjects touched by chosen regulatory documents.

Due to the length of regulatory documents and the complexity of the manual verification, only three regulations were selected to estimate the sentences extraction and attachment of annotations. These regulations refer to different topics, such as customs (32021R0444), external relations (32017R1563), and digital single market (32019R0517), thereby covering different areas despite the small number of test data.

[38] `https://eur-lex.europa.eu/browse/summaries.html`

The evaluation was done manually by first creating the gold standard with definitions and their relations, as well as with definitions and corresponding text segments in CSV format, and then comparing the results to the gold standard.

Evaluation of Information Extraction

In order to draw a conclusion about the general correctness of the implemented prototype, well-known precision, recall, and F-measure used in the field of Information Retrieval can be applied for the evaluation of the Information Extraction task [33]. The aim of the assessment is to prove that the tool achieves an acceptable level of performance and that it represents an improvement over existing approaches. We estimate each phase of extracting information separately and furthest compute the overall scoring of the system by averaging all calculated results.

Definition 5.2.1 (Precision) Given a set of relevant pairs, the precision is given by

$$Precision = \frac{|relevant\ pairs\ \cap\ retrieved\ pairs|}{|retrieved\ pairs|}$$

Definition 5.2.2 (Recall) Given a set of relevant pairs, the recall is given by

$$Recall = \frac{|relevant\ pairs\ \cap\ retrieved\ pairs|}{|relevant\ pairs|}$$

Definition 5.2.3 (F_1) Given the precision and the recall with equal weights, then to calculate the F_1 we can use the equation:

$$F_1 = \frac{Precision * Recall}{0.5 * (Precision + Recall)}$$

Precision demonstrates whether a system is able to output top ranked relevant pairs as per query from the group of various items, while recall indicates the system's potential to discover relevant pairs as per query [34]. Both measures satisfy such qualities as *positiveness*, *completeness-maximality* and *correctness-maximality*. The property of positiveness means the values of semantic measures cannot be negative, while completeness-maximality implies that the value of the recall achieves 1 if all correspondences in relevant pairs can be inferred by retrieved pairs. In the same way, the property of correctness-maximality indicates that the value of the precision is 1 if all correspondences in retrieved pairs can be inferred by relevant pairs. [35] F_1 is the weighted harmonic mean of precision and recall.

The relevance is a foundation on which information extraction evaluation stands, which further can complicate the measurement of the system's effectiveness [34]. To facilitate the understanding of the term, we consider relevant pairs as the pairs that the prototype correctly identified and extracted, while the retrieved pairs refer to the pairs that are manually listed in the gold standard.

Definitions and Relations Extraction

Due to the fact that the extraction of semantic relations is directly dependent on the correctness of extracted definitions, both tasks of definition and relation extraction are evaluated in conjunction. To calculate all the measures, we created gold standard files with manually extracted definitions and corresponding relations as a CSV file and compared them with the resulting values. The Table 1 demonstrates the findings.

Regulation CELEX	**Number of definitions**	**Number of correct extracted definitions**	**Number of relations**	**Number of correct identified relations**
32022R2379	16	16	17	17
32022R0869	19	19	34	30 (4 incorrect)
32021R0947	14	13	18	17
32020R0852	23	23	32	29 (1 incorrect)
32020R0740	24	24	29	28
32019R2144	26	26	30	30
32019R1700	15	15	22	22
32019R1241	50	50	63	61 (2 partly correct)
32019R1154	18	18	18	18
32019R1148	14	14	20	18
32019R1009	25	25	32	31 (1 partly correct)
32019R0833	30	30	39	34 (7 incorrect)
32019R0517	6	6	7	7
32018R1999	62	62	71	71
32018R0848	75	75	85	83 (2 incorrect)
32021R0444	5	5	5	5
32017R1563	4	4	4	4 (1 incorrect)
32016R0679	26	26	42	41

Table 1
Results after evaluating definition and relation extraction tasks.

We listed all the detected issues for each regulations:

- **32022R0869**: problems with definitions *work* (nouns cannot be recognized correctly by scaCy noun_chucks) and *project promoter* ("in the case of"-structure with long enumeration before hyperonym).
- **32021R0947**: the definition *local authority* is not recognized by the tool since it uses *encompass* verb, which is not identified as a synonym of words mean, include, be. As a consequence, one of the relations is missing.
- **32020R0852**: problems with definitions *good status* ("and" is located in the middle of the sentence and, therefore, not all hyperonyms can be identified) and *good condition* (hyperonym is in the middle of the sentence).
- **32019R1241**: hyperonyms of *shore seines* are only partially identified (*nets* instead of *surrounding nets*, *seines* instead of *towed seines*).
- **32019R1148**: the other two hyperonyms of *agricultural activity* are not recognized as nouns by spaCy.
- **32019R1009**: the hyperonym of *placing on the market* is only partly identified (*first making* instead of *first making available*).
- **32019R0833**: problems with definitions *port* (hyperonyms were not identified as nouns), *transhipment* (hyperonym was not labeled as a noun), and *fishing activities* (the listing pattern was not fully recognized).
- **32018R0848**: the definitions *produced by GMOs* and *produced from GMOs* do not have any semantic relations in the explanations in general.
- **32017R1563**: in the definition *beneficiary person* the hyperonym is not the first noun.
- **32016R0679**: one hyperonym of *main establishment* is missing (the listing pattern was not identified).

After examining inaccurate behavior in extracting semantic relations, we figured out that the *noun_chucks* function provided by spaCy often faulty recognized first occurred noun(s) by legal definitions, especially, if a hyperonym represented a verbal noun. Additionally, the location of the hyperonym influenced the accuracy of the approach since we assume that, in most cases, the first detected noun in the explanation has to be a hyperonym. A further wrong behavior lies in treating multiple hyperonyms, namely, the prototype expects a subsequent token of the discovered hyperonym to indicate a listing pattern (e.g., *and*, *or*, comma), although some legal definitions do not adhere to this arrangement.

To estimate the overall performance of the approach we calculate precision and recall as followed:

$$Precision_{overall} = \frac{\sum_{i=1}^{n} Precision_i}{n} = \frac{\sum_{i=1}^{n} \frac{correct\ identified\ relations_i}{all\ relations\ identified_i}}{n}$$

$$Recall_{overall} = \frac{\sum_{i=1}^{n} Recall_i}{n} = \frac{\sum_{i=1}^{n} \frac{correct\ identified\ relations_i}{all\ relations_i}}{n}$$

where n is a number of regulations. The individual values of precision and recall are received from Table 2, where partly correct values are assumed to be incorrect.

Regulation CELEX	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
32022R2379	1	1	1	1	1	1
32022R0869	1	1	1	0.882	0.882	0.882
32021R0947	1	0.929	0.963	1	0.944	0.971
32020R0852	1	1	1	0.967	0.936	
32020R0740	1	1	1	1	0.966	0.983
32019R2144	1	1	1	1	1	1
32019R1700	1	1	1	1	1	1
32019R1241	1	1	1	0.968	0.968	0.968
32019R1154	1	1	1	1	1	1
32019R1148	1	1	1	1	0.9	0.947
32019R1009	1	1	1	0.969	0.969	0.969
32019R0833	1	1	1	0.829	0.872	0.85
32019R0517	1	1	1	1	1	1
32018R1999	1	1	1	1	1	1
32018R0848	1	1	1	0.976	0.976	0.976
32021R0444	1	1	1	1	1	1
32017R1563	1	1	1	0.8	1	0.889
32016R0679	1	1	1	1	0.976	0.989
Total	1	0.996	0.998	0.967	0.967	0.967

Table 2
Precision, recall, and F_1 achieved in definition and relation extraction for each regulation, along with an overall score.

To sum up, in 99.8% of the cases legal terms that the prototype extracted are complete and correct, while in 96.7% of the cases, the system delivers accurate semantic relations among legal definitions.

Sentences Extraction and Assignment of Annotations

After comparing the length and the content of the regulations, we selected the Regulation (EU) 2017/1563 of the European Parliament and of the Council of 13 September 2017 on the cross-border exchange between the Union and third countries of accessible format copies of certain works and other subject matter protected by copyright and related rights for the benefit of persons who are blind, visually impaired or otherwise print-disabled[39], the Regulation (EU) 2019/517 of the European Parliament and of the Council of 19 March 2019 on the implementation and functioning of the .eu top-level domain name and amending and repealing Regulation (EC) No 733/2002 and repealing Commission Regulation (EC) No 874/2004[40], and the Regulation (EU) 2021/444 of the European Parliament and of the Council of 11 March 2021 establishing the Customs programme for cooperation in the field of customs and repealing Regulation (EU) No 1294/2013[41] for the further investigation of extracted sentences including legal definitions and the assigned annotations. Since these two functionalities are implemented together, the evaluation is performed side by side as well.

For this objective, we again created gold standard files in CSV format depicting each legal definition and corresponding sentences. To discover all sentences manually, we used the search function for each term and copied a detected text segment as a sentence. We purposely did not consider article headings since they do not supply relevant information, although the approach did not differentiate between sentences. Therefore, identified headlines regard as irrelevant. The annotations were controlled manually on account of the fact that their existence is easy to spot, as they are highlighted in bright yellow, as illustrated in Figure 3. Pursuing the evaluation, we calculated precision, recall, and F_1 and summarized the results in Table 3.

Regulation CELEX	**Precision**	**Recall**	**F_1**	**Number of sentences**	**Number of correct extracted sentences**
32021R0444	1	1	1	45	45
32017R1563	1	1	1	26	26
32019R0517	1	0.99	0.995	96	95 (6 irrelevant)
Total	1	0.997	0.998		

Table 3
Precision, recall, and F_1 achieved in sentences extraction are presented for each regulation, along with an overall score. Further evaluation data concerning the number of correct extracted sentences is depicted too.

[39] http://data.europa.eu/eli/reg/2017/1563/oj
[40] http://data.europa.eu/eli/reg/2019/517/oj
[41] http://data.europa.eu/eli/reg/2021/444/oj

After estimating the result, we conclude that sentences that were extracted mentioned definitions at least once and, therefore, count as relevant for users. Other than that, in 99.7% of cases, all existing text segments containing legal terms were detected and annotated. Overall, the general performance of the approach lies at 99.8%.

Error Handling of Invalid Input

Intending to demonstrate whether and how the prototype identifies invalid input before analyzing an entered regulation, we selected regulatory documents which did not pass the specified criteria. Furthermore, we investigated non-existent CELEX numbers. Table 4 illustrates the findings together with a format of the examined legal document.

Regulation CELEX	**Format**	**Error message type**
32021R0692	Regulation without *"Definitions"*	The regulation does not contain legal definitions and cannot be processed.
32021R0267	Regulation without *"Definitions"*	The regulation does not contain legal definitions and cannot be processed.
32022L2555	Directive	The legal document has to be a regulation.
32022D1628	Decision	The legal document has to be a regulation.
30022R2555	Invalid Input	The year of a regulation is invalid.
32012R2555	Invalid Input	The entered CELEX does not exist.
52022SC0730	Commission	This type of sector is not supported, please enter a CELEX number of a regulation.

Table 4
Results of evaluating regulatory documents in invalid format and the detailed error message received from the prototype.

Ultimately, the prototype was able to establish all false inputs beforehand and provided a meaningful error message for a user. The attempt to enter a shorter CELEX number was unsuccessful since the tool only supports ten characters by the input.

Functional and Non-functional Requirements Fulfillment

One of the best practices in product development is to differentiate between the baseline functionality essential for the prototype and features that distinguish the approach from similar systems. These features may supplement the main functionality of the prototype or, through including some quality attribute, provide new different functionality. In information systems development, it is advantageous

to stage core functionality for early releases of the system and present new additional features over subsequent releases. [36]

Accordingly, we focus on two kinds of system requirements called functional and non-functional requirements. Functional requirements (FR) depict the functionality of an implemented system, while non-functional requirements (NFR) relate to the quality of service (QoS), e.g., response time, availability, and cost [37]. Both requirements types play an important role in web service composition, and that is why have to be examined to guarantee that the system is user-friendly and meets the quality of service standards.

We formulated individual requirements referring to the implemented prototype:

Functional requirements.

1. The system must be able to accept a valid CELEX number and load a corresponding regulation.
2. The system must be able to identify an invalid CELEX number and stop processing an entered regulation.
3. The system shall allow users to be redirected to the original regulation after analyzing a regulation.
4. The system shall allow users to download a text file with extracted legal definitions.
5. The system shall allow users to download a text file with text segments containing definitions.
6. The system shall allow users to download a text file with identified semantic relations among legal definitions.
7. The system shall allow users to download a modified HTML regulatory document with annotations.

Non-functional or quality requirements.

1. The system shall send users a specific error message in case of mismatch, describing a current issue.
2. The system shall process regulations only in English language.
3. The system shall be able to process lengthy regulations.
4. The system shall produce notable annotations in a modified regulation.

5. The system shall produce readable and easy-to-understand text files so that users can rapidly find the required information.
6. The system shall list the articles in which the definitions are located.
7. The system shall demonstrate the overall number of extracted legal terms.
8. The system shall demonstrate first five most frequent legal definitions to users.
9. The system shall provide a tutorial for new users in order to simplify the understanding of how to use the tool.

The functional requirements were evaluated using the regulations provided in Section 5.2. All seven points appeared to be fulfilled by all legal documents, including invalid inputs. Moving to the non-functional requirements, which describe additional features of the prototype, we gradually validate adherence to the requirements.

Beginning with NFR1, the requirement is accomplished, as depicted in 5.2. The system always processes regulations in English (NFR2) since the CELEX number is independent of the language, and the modified URL redirects to the English version of a legal document without exception. The variety of regulatory documents with different lengths (NFR3) was evaluated in Section 5.2, including lengthy regulations (e.g., 32018R0848[42]). While estimating FR7, NFR4 was investigated in parallel, and highlighted with yellow annotations were easy to distinguish from the rest of the legal text. For the accomplishment of NFR5, the prototype uses various sorting techniques depending on the task. Figure 2 illustrates the segments of the resulting text files after processing a Regulation (EU) 2017/1563 of the European Parliament and of the Council of 13 September 2017[43]. As evident from the presented results, the text file containing pure legal definitions lists the terms in the order they are mentioned in the regulation. Simultaneously, in the document with extracted text segments, the tool differentiates between definitions and itemizes all sentences referring to them alongside the list of articles where they occur (NFR6). This way, the user can directly search for a specific definition and find relevant information, which makes text files easy to understand. Due to the extensive number of text segments, the readability factor is violated. Lastly, a text file visualizing identified semantic relations is sorted alphabetically and, therefore, facilitates the search of a specific instance. Besides, it depicts a hyponymy tree at the end of the file for enhanced visualization. The

[42] http://data.europa.eu/eli/reg/2018/848/oj
[43] http://data.europa.eu/eli/reg/2017/1563/oj

subsequent quality requirements NFR7 and NFR8 showed to be achieved after processing each regulation from the testing data set, while the last NFR9 is not fulfilled yet.

Text file with legal definitions

```
(1)‘work or other subject matter’ means a work in the form of a
book, journal, newspaper, magazine or other kind of writing,
notation, including sheet music, and related illustrations, in any
media, including in audio form such as audiobooks and in digital
format, which is protected by copyright or related rights and which
is published or otherwise lawfully made publicly available;
(2)‘beneficiary person’ means, regardless of any other disabilities,
a person who:(a)is blind;(b)has a visual impairment which cannot be
improved so as to give the person visual function substantially
```

Text file with text segments

```
Definition: beneficiary person
Total number of text segments including definition: 13

Definition beneficiary person can be found in: Article 2; Article 3;
Article 4; Article 5;

                Since the objective of this Regulation, namely to
implement in a uniform manner the obligations under the Marrakesh
Treaty with respect to the export and import between the Union and
third countries that are parties to the Marrakesh Treaty of
accessible format copies of certain works and other subject matter
for non-commercial purposes for the benefit of beneficiary persons,
and to lay down the conditions for such export and import, cannot be
sufficiently achieved by the Member States, but can rather, by
reason of its scale and effects, be better achieved at Union level,
the Union may adopt measures in accordance with the principle of
subsidiarity as set out in Article 5 of the Treaty on European
Union. In accordance with the principle of proportionality as set
out in that Article, this Regulation does not go beyond what is
necessary in order to achieve that objective.
```

Text file with semantic relations

```
accessible format copy is a hyponym of copy
authorised entity established in a Member State is a hyponym of
entity
beneficiary person is a hyponym of other disabilities
beneficiary person is a hyponym of person
work or other subject matter is a hyponym of work

Hyponymy Tree:
other disabilities
| beneficiary person
entity
| authorised entity established in a Member State
copy
| accessible format copy
person
| beneficiary person
work
| work or other subject matter
```

Figure 2
Segments of three resulting text files containing legal definitions, text segments, and semantic relations after processing the regulation on the cross-border exchange between the Union and third countries.

After analyzing the evaluation results, we can draw the conclusion that all of the functional and most of the non-functional requirements were accomplished, and thereby, the tool should be able to function as intended and provide a satisfactory user experience. Although, to fully evaluate the efficiency of the prototype and identify any potential areas for improvement, further evaluation methods and feedback from users is necessary.

Discussion

Addressing the issues referring to manual analysis of regulatory documents and common misinterpretation of legal text, the introduced approach was able to identify and extract legal definitions in 99.8% of cases as well as their semantic relations in 96.7% of cases, such as hyponymy, meronymy, and synonymy, using NLP techniques, particularly dependency parsing, tokenization, and POS-tagging. The obtained data were collected in separate files based on the solving task and presented to users on the implemented web service. Furthermore, we visualized valuable statistics like the frequency of terms in text segments as well as their locations, so the users can draw conclusions about specific definitions on their own and additionally assume the objective of the regulation before reading it.

Evaluating the findings of definition extraction, the approach achieved excellent results in detecting the article containing legal terms and extracting each of them, removing all redundant information as enumeration. Similar to the existing methods, we applied pattern matching for relation extraction, which appeared successful for the selected data set. However, with an increased amount of testing data, the amount of faulty detected relations can also increase and influence the approach's performance if some patterns are missing. The assumption that each definition contains at least one semantic relation turned out to be incorrect for highly unusual cases when the explanation described the meaning of the term without referring to any noun but more to the whole process. We visualized the obtained information by outputting the results as text or HTML files, where the modified regulation contains annotations with legal terms' explanations, reducing the misinterpretation of the legal text. In contrast, the text file with semantic relations reveals the interconnections and helps match the definitions' meanings. More valuable data concerning the regulation (title and date of execution) and legal definitions (number of legal definitions, occurrence, and position) was further depicted on the resulting page, accomplishing the last research question.

Most existing approaches pointed to a similar structure of regulatory documents and sentences specifying definitions [5], [18], [23]. The results confirmed this theory by applying rule-based techniques in order to extract legal definitions. While previous research has focused chiefly on identifying legal terms from the whole document and occasionally building an ontology or investigating hyponymy, the approach demonstrated good results by obtaining legal definitions previously defined in the

separate article and examining their semantic relations, further concentrating on meronymy and synonymy. Moreover, the prototype offered a valuable visualization of the extracted information by attaching the explanations to the legal terms found in regulations and providing the resulting documents to users.

Limitations

Nonetheless, the introduced approach for definition and relation extraction has its limitations. One of the issues is that a legal definition can use another sentence root which does not belong to the synonyms of words "mean", "include", and "be" but still can indicate a valid legal term containing semantic relations. In this case, the prototype faulty ignores the definition, and since, in general, any incorrect behaviour in definition extraction directly impacts the results of relation extraction, the tool does not proceed with analyzing possible relations.

For extracting semantic relations, detecting noun chunks has proven to be quite successful, although the prototype faultily neglected some verbal nouns, which led to detecting wrong relations. A further limitation in relation extraction is a listing pattern, where the noun chunks do not directly follow each other but are distributed along the whole text segment. Applying dependency parsing and using the structural properties of compound words, as presented in [38], did not solve any of the listed issues.

Future Work

Reflecting on the research findings, we discuss the possible directions for further investigation and propose potential concepts that may build on the current approach. Since the prototype concentrates only on regulations, the approach may be extended to decisions and directives, which include an article listing legal definitions. Besides, the focus only on regulatory documents containing this specific article can be changed to the legal acts, which particularly lack a section defining the terms. In this case, the developed prototype can search for possible legal definitions in the whole document and collect them in one article.

A further intriguing aspect for analyzing in future work is the homogeneity of text segments, i.e. whether the whole segment primarily refers to a single definition or multiple terms occur in the text. In this case, the prototype should compare the number of occurrences of each definition in the text segment and decide about the homogeneity detecting the predominance of a particular term.

Concerning visualization, diverse semantic relations can be collected in a graph contingent upon a particular definition. The general graph illustrating all legal definitions and their relations in most cases can present difficulties for interpretation due to its intricacy. Besides, through expert interviews, the structure of the text file containing extracted text segments can be adjusted to increase readability, as well as more relevant concepts can be emphasized and obtained for the users. To enhance the representation of annotations, it is feasible to highlight all legal definitions mentioned in a text after hovering over the initial reference of the term. This way, readability will remain unaffected while the users can virtually identify and observe each term.

In conclusion, future research can focus on improving the current approach based on the detected limitations and further investigate the semantic relation patterns in order to increase the overall precision.

Conclusion

This research aimed to develop a practical approach for semi-automating the procedure of regulation analysis, intending to accelerate the investigation and reduce the misinterpretation of legal text. Applying pattern matching and NLP techniques for definition and relation extraction, we received excellent results in achieving the objectives. We concluded that legal definitions and their semantic relations are essential factors to consider when studying regulatory documents. Visualizing these aspects proves advantageous for users who seek to obtain a brief understanding of the regulation's scope and objective, along with simplifying the legal search for relevant information.

Starting with analyzing the structure of regulatory documents, we established that legal terms are commonly defined in a corresponding article for improving general perception. To reduce the manual searching time for this article, the approach lists occurring legal definitions and their respective frequencies and locations. Moreover, it identifies semantic relations among them and presents a catalogue of text segments which mention each term. By utilizing the prototype, users minimize time spent attempting to understand the content of the regulation and can concentrate only on the relevant definitions. In cases where comprehensive document processing is required, the tool offers a modified regulation containing annotations with explanations so the users do not need to verify the meaning of the appearing terms again. The evaluation of the approach using information retrieval metrics such as precision, recall, and F-measure, as well as investigating the fulfilment of requirements, showed acceptable results for all addressed tasks. Most of the incorrect results arose from the complex structure of sentences or the utilization of verbal nouns in explanation.

However, future research can further enhance the proposed prototype, e.g. in the visualization aspects of depicting text segments mentioning legal definitions and illustrating semantic relations. Expert interviews can benefit to determine in which way presenting of obtained information is desired and which additional data concerning the content of regulations the approach should consider.

The introduced prototype for automatic extraction of legal definitions and semantic relations from European regulatory documents significantly contributes to improving the accessibility and understanding of the legal text. The research successfully addressed the legal information retrieval task, presenting a solution to the manual processing of regulatory documents, further focusing on the

research gap of inaccurate misinterpretation of legal text due to domain-specific language. After studying various regulations belonging to a wide range of topics, we confirmed the assumption of structural similarity of defining legal terms and, thereby, received accurate results in identifying definitions and semantic relations among them, offering a deeper understanding of their meaning.

Bibliography

[1] E. Ferneda, H. A. do Prado, A. H. Batista, and M. S. Pinheiro, "Extracting definitions from brazilian legal texts," in *Computational Science and Its Applications - ICCSA 2012 - 12th International Conference, Salvador de Bahia, Brazil, June 18-21, 2012, Proceedings, Part III*, B. Murgante, O. Gervasi, S. Misra, *et al.*, Eds., ser. Lecture Notes in Computer Science, vol. 7335, Springer, 2012, pp. 631–646. DOI: 10.1007/978-3-642-31137-6_48. [Online]. Available: https://doi.org/10.1007/978-3-642-31137-6%5C_48.

[2] D. Jurafsky and J. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Jan. 2023, vol. 3.

[3] T. Seyffarth and S. Kühnel, "Maintaining business process compliance despite changes: A decision support approach based on process adaptations," *J. Decis. Syst.*, vol. 31, no. 3, pp. 305–335, 2022. DOI: 10.1080/12460125.2020.1861920. [Online]. Available: https://doi.org/10.1080/12460125.2020.1861920.

[4] M. Hashmi, G. Governatori, H.-P. Lam, and M. T. Wynn, "Are we done with business process compliance: State of the art and challenges ahead," *Knowledge and Information Systems*, vol. 57, no. 1, pp. 79–133, 2018.

[5] B. Amaludin, F. R. Wardika, P. J. M. Putra, and I. G. Y. Paramartha, "Analyze the usage of legal definitions in indonesian regulation using text mining case study: Treasury and budget law," in *Legal Knowledge and Information Systems - JURIX 2021: The Thirty-fourth Annual Conference, Vilnius, Lithuania, 8-10 December 2021*, S. Erich, Ed., ser. Frontiers in Artificial Intelligence and Applications, vol. 346, IOS Press, 2021, pp. 107–112. DOI: 10.3233/FAIA210324. [Online]. Available: https://doi.org/10.3233/FAIA210324.

[6] C. Sansone and G. Sperli, "Legal information retrieval systems: State-of-the-art and open issues," *Inf. Syst.*, vol. 106, p. 101 967, 2022. DOI: 10.1016/j.is.2021.101967. [Online]. Available: https://doi.org/10.1016/j.is.2021.101967.

[7] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, pp. 75–105, 2004.

[8] A. Kao and S. R. Poteet, *Natural language processing and text mining*. Springer Science & Business Media, 2007.

[9] G. A. Al-Talib and A. A. Atiyah, "Extraction and classification of semantic relations from news recommendation," in *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, IEEE, 2022, pp. 1–3.

[10] J. Nivre, "Dependency parsing," *Language and Linguistics Compass*, vol. 4, no. 3, pp. 138–152, 2010.

[11] M. Le Nguyen, H. T. Nguyen, P.-T. Nguyen, T.-B. Ho, and A. Shimazu, "An empirical study of vietnamese noun phrase chunking with discriminative sequence models," in *Proceedings of the 7th Workshop on Asian Language Resources (ALR7)*, 2009, pp. 9–16.

[12] S. Ibrihich, A. Oussous, O. Ibrihich, and M. Esghir, "A review on recent research in information retrieval," *Procedia Computer Science*, vol. 201, pp. 777–782, 2022.

[13] M. Gomes, B. Oliveira, and C. Sousa, "Enriching legal knowledge through intelligent information retrieval techniques: A review," in *Progress in Artificial Intelligence - 21st EPIA Conference on Artificial Intelligence, EPIA 2022, Lisbon, Portugal, August 31 - September 2, 2022, Proceedings*, G. Marreiros, B. Martins, A. Paiva, B. Ribeiro, and A. Sardinha, Eds., ser. Lecture Notes in Computer Science, vol. 13566, Springer, 2022, pp. 119–130. DOI: 10.1007/978-3-031-16474-3_11. [Online]. Available: https://doi.org/10.1007/978-3-031-16474-3%5C_11.

[14] D. Locke and G. Zuccon, "Case law retrieval: Problems, methods, challenges and evaluations in the last 20 years," *arXiv preprint arXiv:2202.07209*, 2022.

[15] C. Randier, "Definitions for harmonising legal terminology," *Harmonising Legal terminology, Bolzano, Publication EURAC Research*, pp. 91–105, 2008.

[16] E. Ferneda, H. A. do Prado, A. H. Batista, and M. S. Pinheiro, "Extracting definitions from brazilian legal texts," in *Computational Science and Its Applications - ICCSA 2012 - 12th International Conference, Salvador de Bahia, Brazil, June 18-21, 2012, Proceedings, Part III*, B. Murgante, O. Gervasi, S. Misra, *et al.*, Eds., ser. Lecture Notes in Computer Science, vol. 7335, Springer, 2012, pp. 631–646. DOI: 10.1007/978-3-642-31137-6_48. [Online]. Available: https://doi.org/10.1007/978-3-642-31137-6%5C_48.

[17] R.-H. Hwang, Y.-L. Hsueh, and Y.-T. Chang, "Building a taiwan law ontology based on automatic legal definition extraction," *Applied System Innovation*, vol. 1, no. 3, p. 22, 2018.

[18] M. Nakamura, Y. Ogawa, and K. Toyama, "Extraction of legal definitions from a japanese statutory corpus-toward construction of a legal term ontology," in *LAW VIA THE INTERNET CONFERENCE*, 2013, pp. 1–11.

[19] S. Höfler, A. Bünzli, and K. Sugisaki, "Detecting legal definitions for automated style checking in draft laws," *Technical Reports in Computational Linguistics*, no. CL-2011.01, 2011.

[20] S. Walter, "Definition extraction from court decisions using computational linguistic technology," *Formal Linguistics and Law*, vol. 212, p. 183, 2009.

[21] E. De Maat, R. Winkels, and T. van Engers, *Making sense of legal texts*. Walter de Gruyter, 2009, vol. 212.

[22] B. Waltl, J. Landthaler, E. Scepankova, *et al.*, "Automated extraction of semantic information from german legal documents," in *IRIS: Internationales Rechtsinformatik Symposium*, 2017.

[23] E. de Maat and R. Winkels, *Automated classification of norms in sources of law*. Springer, 2010.

[24] A. Korger and J. Baumeister, "Rule-based semantic relation extraction in regulatory documents.," in *LWDA*, 2021, pp. 26–37.

[25] G. Boella, L. Di Caro, and L. Robaldo, "Semantic relation extraction from legislative text using generalized syntactic dependencies and support vector machines," in *Theory, Practice, and Applications of Rules on the Web: 7th International Symposium, RuleML 2013, Seattle, WA, USA, July 11-13, 2013. Proceedings 7*, Springer, 2013, pp. 218–225.

[26] K. Fundel, R. Küffner, and R. Zimmer, "Relex—relation extraction using dependency parse trees," *Bioinformatics*, vol. 23, no. 3, pp. 365–371, 2007.

[27] M. Garcia, "Semantic relation extraction. resources, tools and strategies," in *International Conference on Computational Processing of the Portuguese Language*, Springer, 2016, pp. 141–152.

[28] A. Schutz and P. Buitelaar, "Relext: A tool for relation extraction from text in ontology extension," in *International semantic web conference*, Springer, 2005, pp. 593–606.

[29] E. Commission and L. service, *Joint practical guide of the European Parliament, the Council and the Commission for persons involved in the drafting of European Union legislation*. Publications Office, 2016. DOI: doi/10.2880/5575.

[30] D. A. Cruse, "Hyponymy and its varieties," *The semantics of relationships: an interdisciplinary perspective*, pp. 3–21, 2002.

[31] C. S. Khoo and J.-C. Na, "Semantic relations in information science," *Annu. Rev. Inf. Sci. Technol.*, vol. 40, no. 1, pp. 157–228, 2006.

[32] V. Lyytikäinen, P. Tiitinen, A. Salminen, L. Mercier, and J.-L. Vidick, "Visualizing legal systems for information retrieval.," in *IRMA Conference*, Citeseer, 2000, pp. 245–249.

[33] D. Maynard, W. Peters, and Y. Li, "Metrics for evaluation of ontology-based information extraction.," in *EON@ WWW*, 2006.

[34] K. Zuva and T. Zuva, "Evaluation of information retrieval systems," *International journal of computer science & information technology*, vol. 4, no. 3, p. 35, 2012.

[35] Q. Ji, Z. Gao, Z. Huang, and M. Zhu, "Semantic precision and recall for evaluating incoherent ontology mappings," in *Active Media Technology - 8th International Conference, AMT 2012, Macau, China, December 4-7, 2012. Proceedings*, R. Huang, A. A. Ghorbani, G. Pasi, T. Yamaguchi, N. Y. Yen, and B. Jin, Eds., ser. Lecture Notes in Computer Science, vol. 7669, Springer, 2012, pp. 338–347. DOI: 10.1007/978-3-642-35236-2_34. [Online]. Available: https://doi.org/10.1007/978-3-642-35236-2%5C_34.

[36] R. Malan, D. Bredemeyer, *et al.*, "Functional requirements and use cases," *Bredemeyer Consulting*, 2001.

[37] M. Chen, T. H. Tan, J. Sun, Y. Liu, J. Pang, and X. Li, "Verification of functional and non-functional requirements of web service composition," in *Formal Methods and Software Engineering: 15th International Conference on Formal Engineering Methods, ICFEM 2013, Queenstown, New Zealand, October 29–November 1, 2013, Proceedings 15*, Springer, 2013, pp. 313–328.

[38] A. Hippisley, D. Cheng, and K. Ahmad, "The head-modifier principle and multilingual term extraction," *Natural Language Engineering*, vol. 11, no. 2, pp. 129–157, 2005.